PhD thesis

Automatic abstractive summarization of long medical texts with multi-encoders Transformer and general-domain summary evaluation with wikiSERA

Thèse de doctorat de l'Université Sorbonne Paris Nord préparée au Laboratoire d'Analyse Sémantique Texte et Image (LASTI) et au Laboratoire d'Informatique de Paris-Nord (LIPN)

Spécialité du doctorat: Informatique

Thèse présentée et soutenue à l'université Sorbonne Paris Nord, le 5 Mai 2021, par

Jessica LÓPEZ ESPEJEL

Composition du jury:

Xavier TANNIER Professeur, Sorbonne Université	Président
Juan Manuel TORRES MORENO Maître de conférences HDR, Université d'Avignon	Rapporteur
Benoit FAVRE Maître de conférences HDR, Université d'Aix-Marseille	Rapporteur
Nathalie PERNELLE Professeur, Université Sorbonne Paris Nord	Examinatrice
Thierry CHARNOIS Professeur, Université Sorbonne Paris Nord	Directeur de thèse
Gaël DE CHALENDAR Ingénieur Chercheur, CEA-LIST	Encadrant scientifique
Jorge GARCÍA FLORES Docteur, Université Sorbonne Paris Nord	Coencadrant scientifique
Iván Vladimir MEZA RUÍZ Docteur, Université Autonome du Mexique (IIMAS)	Coencadrant scientifique

Contents

List of Figures

Chapter 1

Introduction

1.1 Context

Due to the enormous information volume and the continuous interest in research on various health diseases, the number of medical articles has been increasing over the years. Nowadays, thousands of institutions and researchers are held to tackle the Coronavirus (Covid-19) and handle the pandemic situation. We present hereafter a list of some statistics from over the world that show the huge increase in digital medical data specifically:

- In 2020, the number of submissions to Elsevier's journals increased by 58% between February and May compared to the same period in 2019 (Squazzoni et al., 2020).

- The number of health articles increased by 92% in 2020, where scientists published more than 100,000 articles about the Covid-19 (nature, 2020).

- According to the EMC Digital Universe with Research and Analysis by IDC[1], there was an enormous growth in the global healthcare data between 2013 and 2020. Figure 1.1 shows the difference in data volume between the two years in exabytes.

- In the first year of its existence, the US National Cancer Institute[2] received between 2016 and 2017 over 4.5 petabytes of data from research institutions.

At present, web sites such as PubMed (for Biotechnology Information, 2018) from MEDLINE and Dimensions (Solutions, 2021) contain millions of medical texts coming from different sources such as books, life science journals, and articles. Consequently, it is not possible for a human to read all this information promptly.

[1] https://www.idc.com/

[2] https://www.cancer.gov/news-events/cancer-currents-blog/2017/gdc-dave-tools

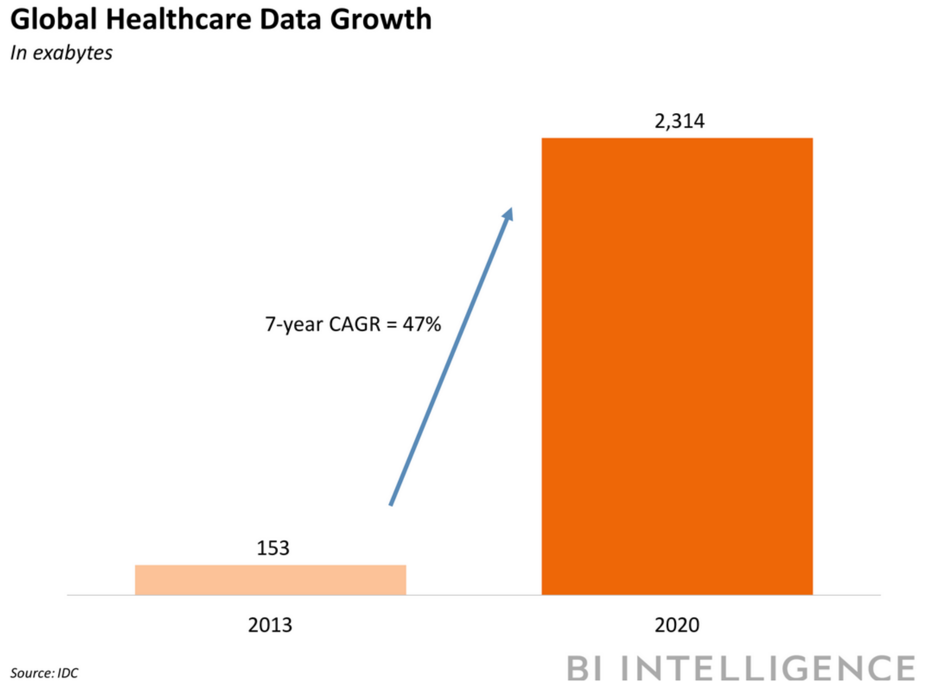


Figure 1.1: The growth in global healthcare data between 2013 and 2020

In order to keep up with the rapid progress in the medical domain, doctors and researchers need to quickly extract relevant information from medical articles to further develop their research and save more lives. Fortunately, artificial intelligence advancements make this task feasible with the emergence of Automatic Text Summarization (ATS). The ATS is an active research area in Natural Language Processing (NLP) whose objective is to automatically produce a summary concentrating the most important information from a long source document or a document collection (Mani, 2001).

The first summarization approaches were extractive, where a summary is built by identifying relevant pieces of text and concatenating them. The most popular methods include frequency-based techniques, probabilistic models, and machine learning. Later, the scientific community moved the research along to abstractive approaches with the use of deep learning. Here, a summary is built by paraphrasing the text in a readable and consistent short paragraph.

Despite the evolution in automatic text summarization, an effort is also needed to automatically assess the quality of generated summaries and thus be able to compare and improve different ATS systems. Human evaluation is the best reference to evaluate summaries. However, such a process is expensive in terms of time, money, and effort. Therefore, the scientific community has developed various extrinsic and intrinsic methods to evaluate summaries automatically (Jones and Galliers, 1996). In an extrinsic evaluation, summaries are assessed within the context of another task, like answer extraction. In an intrinsic evaluation, summaries are assessed outside a context, with or without human intervention. Both extrinsic and intrinsic techniques aim to evaluate some characteristics in the summaries, such as linguistic quality, content, coherence, and coverage.

In this thesis, we tackle both automatic summarization and automatic evaluation of summaries. On the one hand, we focus on abstractive summarization that is closer to how humans write summaries (by understanding the main idea of a text and then rephrasing it differently). On the other hand, we focus on intrinsic methods to assess the quality of abstractive summaries that belong to the general domain while partially relying on human intervention.

1.2 Challenges

Before developing an automatic summarization or evaluation system, many factors should be taken into consideration. First, the source of evaluation texts: digital documents can be either absorbed from the web, downloaded from public benchmarks, or automatically transcribed from an audio source. Consequently, ethical issues arise regarding the possibility of using these texts without violating the privacy of concerned parties. Second, the nature of evaluation texts: documents can belong to different domains such as medicine, news, sports, literature, science, and dialogues. Consequently, the adequate automatic system is chosen depending on the nature of texts to summarize, their structure, and length. For instance, the maximum input sequence length and the maximum length of generated summaries change from one system to another.

The goal of this thesis is to develop an automatic summarization system that can handle long input sequences and an automatic evaluation approach to assess the quality of generated summaries. However, many challenges arise along with text summarization and summary evaluation. We present some of them in the following two subsections.

1.2.1 Automatic Text Summarization

In this work, we are interested in the summarization of long medical texts. For this reason, we adopt deep learning techniques for their ability to generate abstractive summaries from long input sequences. Many deep architectures achieve state-of-the-art results in various NLP tasks, such as BERT (Devlin et al., 2019), T5 (Raffel et al., 2020), and PEGASUS (Zhang et al., 2020a). These models are adaptable for text summarization, but suffer from some limitations related to the complexity of the summarization task:

- *Length of input text* - Unfortunately, none of the existing neural-network-based approaches can read the whole source text for memory explosion issues. To the best of our knowledge, the maximum input length in literature is 2000 tokens. It was used by an LSTM-based approach (Cohan et al., 2018) and also by our proposed approach in this thesis.

- *Redundant information* - This is one of the main drawbacks of existing summarization approaches, where generated summaries comprise lots of repetitions. This issue necessitates efficient techniques to avoid repeated n-grams at the decoding level.

- *Choice of output summary* - At the decoding stage, the probability of the next word to predict is based on what was already generated. There are many ways to predict the next word, either by performing a greedy search (where each time the word with the highest probability is chosen) or using more sophisticated searching algorithms such as beam search (where a tree of possible summaries is explored).

- *Computational requirements* - Unlike many NLP applications, text summarization is a difficult task that needs deep networks in order to learn effectively. The best state-of-the-art results were obtained with pre-trained models. For instance, the PEGASUS (Zhang et al., 2020a) system from Google was pre-trained on 1.5 billion articles (of size 3.8 TB). Thus, it is necessary to have powerful memory and computational resources in order to tackle summarization effectively.

- *Numerical data* - One of the major issues related to the summarization of medical articles is the strong presence of numbers such as medicines concentration, patients' age, statistics, quantities, and dates. Since the vocabulary used to train the summarization model is limited (containing the most frequent terms only), it is hard to retain knowledge about all used numbers and invoke them correctly in generated summaries. However, this is a serious problem because the information presented in medical articles is sensitive and should be as precise as possible.

- *Choice of tokenizer* - The role of a tokenizer is to transform a text into a list of tokens. Depending on each tokenizer, only the most important words are kept, which could influence the quality of generated summaries. Many questions arise when cleaning the text before choosing the most convenient tokenizer. For instance: should we keep the numbers? The linking words? Or should we simply separate text tokens based on spaces?

1.2.2 Automatic Summary Evaluation

Despite the difficulties related to Automatic Text Summarization, many systems were developed in the last decade to handle the problem (Zhang et al., 2020a, Cohan et al., 2018, See et al., 2017). However, it is crucial to assess the quality of generated summaries in order to be able to improve automatic summarization systems. Therefore, the Automatic Summary Evaluation domain arises along with Text Summarization to judge if the summaries generated automatically are compact, meaningful, and/or coherent.

Over time, researchers have proposed several automatic systems that facilitate the evaluation of generated summaries. However, these systems tackle many challenges at once, notably:

- *Indeterminism* - In automatic evaluation, there is no “ideal” summary and no unique “correct” one. Summaries can be evaluated based on many criteria, such as their quality, informativeness, and impact on efficiency. Based on each criterion, an evaluation metric can be useful or not (Mitkov, 2004). Also, the evaluation quality does not depend only on the automatic system but also on human competence (in the case where human judgment is mandatory).

- *Infairness* - When an evaluation approach is based on lexical content, it becomes unfair to evaluate abstractive summaries (Lin, 2004). The latter ones do not necessarily copy words from the original text but rather paraphrase it using, for example, synonyms and different linguistic forms.

- *Dependency* - Most of the current abstractive evaluation approaches depend on human reference summaries (also called gold-standards) (Lin, 2004, Cohan and Goharian, 2016), where evaluation is made by comparing a candidate summary with many reference summaries. Some researchers worked without the need for human intervention (Cabrera-Diego and Torres-Moreno, 2018, Torres-Moreno et al., 2010). However, the correlation with manual methods becomes low in such cases.

- *Evaluation domain* - The performance of each system depends on the domain to which candidate summaries belong. For example, some approaches are effective for the biomedical domain (Cohan and Goharian, 2016), while others are more accurate for the news domain (Cabrera-Diego and Torres-Moreno, 2018). Our proposed approach is more suitable to evaluate summaries from the general domain.

Since the automatic text summarization and the automatic summary evaluation cannot be dissociated, the challenges are heavier since there are many aspects to handle in order to provide a summarization system that is as accurate as possible.

1.3 Contributions overview

The main contribution of this thesis is the design and development of an automatic abstractive summarization system of long medical texts (called *HazPi* below). To evaluate such a system, we need an efficient evaluation approach that provides a reasonable estimation of the quality of generated summaries. At the beginning of the thesis, the most popular evaluation approach was ROUGE (Lin, 2004). Unfortunately, this method is biased by lexical similarities between candidate and reference summaries, making it unfair to evaluate abstractive summaries that most probably contain words that exist neither in the original text nor in the reference summaries. This problem was solved later by SERA (*Summarization Evaluation by Relevance Analysis*) (Cohan and Goharian, 2016), which is based on a relevance analysis to evaluate both extractive and abstractive summaries fairly. However, it was designed to be usable in a specialized biomedical domain only. For this reason, it achieved higher correlations than ROUGE for the biomedical domain.

We departed from this motivation and decided that it is annoying to use an evaluation method specific to one domain, especially if future researchers want to use *HazPi* to summarize texts from other domains than medical. It would be more interesting to have an evaluation method that is usable in a general domain. For this reason, our second contribution is an improvement of the SERA approach that we call wikiSERA. wikiSERA is an efficient adaptation of SERA to the automatic evaluation of abstractive summaries that belong to the general domain. We present in this section a brief overview of our contributions in both summarization and evaluation domains.

1.3.1 Automatic Text Summarization

To tackle the text summarization task, we use deep learning models, more precisely Transformers Neural Networks (Vaswani et al., 2017). However, two main problems occur with the ATS task in deep architectures: the first one is the training time model. Depending on the number of available GPUs, this phase could take from few days to few weeks, even with transformers' ability to process sequential input parallelly. The second one is the size of the encoded document (i.e., input sequence length). With the birth of deep learning, the length of the input sequence fed into the model has been significantly reduced compared to extractive predecessor systems from the 2000s (García Flores et al., 2009). For example, some LSTM-based summarization systems (Cohan et al., 2018, See et al., 2017) truncate the source document to 2000 and 400 tokens, respectively. Alternatively, the maximum input length for a Transformer-based NTS system is 1024 tokens (Zhang et al., 2020a). However, scientific articles are much larger. For instance, the biomedical dataset built by Cohan et al. (2018) has, on average, 3016 tokens in each article. To handle these problems, we propose two improvements to the original Transformer model that allow a faster training of the network while increasing the input document size for summarization without penalizing the quality of generated summaries.

Our approach (called *HazPi*) consists of two stages. In the first stage, we propose to modify the transformer neural network architecture to read longer sequences of text by using four input encoders instead of one. Contrarily to existing Transformer-based approaches that achieve high performance by reading up to 512 input tokens, *HazPi* can read up to 2000 tokens of the input text while improving summary quality and reducing execution time by more than half. The sequence size used by our system is closer to the average document length of 3016 tokens of biomedical articles (Cohan et al., 2018).

In the second stage, we propose an extra-training phase (called *End-Chunk Task Training*) inspired by the end-task training from Hoang et al. (2019). Instead of presenting the whole reference summary to the decoder, we feed chunks of summary tokens progressively until consuming the whole sequence.

Finally, we conduct experiments where we pre-train our modified transformer neural network using a large medical dataset. We build this dataset (called CovMed) by combining medical articles from PubMed (for Biotechnology Information, 2018) corpus and Kaggle's Covid-19 dataset (House, 2020).

The code of our approach is available in the following GitHub repository: `https://github.com/JessicaLopezEspejel/HazPi/`.

1.3.2 Automatic Summary Evaluation

As mentioned above, developing an automatic summarization system is an interesting but not easy task. To successfully evaluate the performance of $HazPi$, we need to have a robust automatic evaluation approach well suited for abstractive summarization. ROUGE (Lin, 2004) is one of the most famous automatic approaches that rely on human reference summaries. This metric provides a high correlation with manual methods, mainly in extractive summarization. However, ROUGE is not fair when evaluating abstractive summaries since it is based on lexical overlaps between tokens and phrases in the reference summary and the generated one (Cohan and Goharian, 2016, Lu and Jin, 2020). To overcome this issue, SERA (Cohan and Goharian, 2016) was proposed as an alternative to evaluating abstractive summaries in the biomedical domain. SERA focuses on the semantic content of documents through an information retrieval approach, leading to efficiently assessing the quality of summaries that are lexically different but express the same idea.

The SERA approach is based on a search engine to establish a content-relevance analysis between a candidate summary generated by an ATS system and reference summaries written by humans. SERA uses as an input to the search engine both candidate and reference summaries and a pool of documents that constitute the index. The role of the search engine is to search the queries in the index and return a list of ranked documents based on their similarities with input summaries. Afterward, a score is attributed to the candidate summary based on an intersection between the two sets of documents related to the queries. Unfortunately, despite the effectiveness of SERA in abstractive summary evaluation, it is focused on the biomedical domain only, making it so restrictive to evaluate summaries from other domains. We decided to study SERA and adapt it to evaluate summaries from the general domain by proposing a generic metric that we called wikiSERA in Chapter 3. wikiSERA is helpful to evaluate automatic summarization systems in the medical domain and other domains.

To analyze SERA in more detail, we conduct a POS Tag study on many corpora belonging to different domains: PubMed (biomedical dataset (Cohan et al., 2018)), AQUAINT-2 (a dataset specialized in news (Graff, 2002)), and Wikipedia (general-domain dataset). Our study confirmed the observation of Kieuvongngam et al. (2020) regarding the fact that, in generated summaries, nouns represent more accurately the information conveyed by the original abstracts than other POS tags. We also noticed that percentages of verbs and adjectives are higher in AQUAINT-2 (news) and Wikipedia (general domain) than in the PubMed dataset.

Based on the POS Tag study described above, we propose wikiSERA, an improved version of SERA. We redefine query reformulation by considering nouns, verbs, and adjectives as inputs to the search engine. This query reformulation helps take out SERA from evaluating biomedical summaries to evaluating summaries from the general domain.

We conduct extensive experiments to assess the merits and limitations of SERA, wikiSERA, as well as many influential evaluation approaches from the literature. Results show that wikiSERA achieves competitive results compared to SERA while outperforming in some cases ROUGE, the lexical-based evaluation approach.

The main contributions are:

1. We re-implement SERA from scratch and propose wikiSERA, an improved version of SERA that is domain-independent.

2. We conduct extensive experiments with two large corpora: AQUAINT-2 (news corpus) and Wikipedia (general domain corpus). We compare wikiSERA against several state-of-the-art approaches and provide a comprehensive study of our experiments. Results show the effectiveness of our approach.

3. We make the code and Wikipedia dataset publicly available to facilitate future research. Note that AQUAINT-2 is not open source, and we cannot distribute it. However, obtained results could be helpful in academic research.

The code and data of our approach are available in the following GitHub repository: `https://github.com/JessicaLopezEspejel/wikiSERA/`.

1.4 Thesis plan

This work is organized as follows:

- *Chapter 2*. First, we explain in detail the difference between different categories of automatic text summarization approaches (extractive vs. abstractive, mono-document vs. multi-document, generic vs. query-based). Second, we describe the most popular datasets used by the scientific community for automatic text summarization and summary evaluation. Third, we present the state-of-the-art approaches while trying to make room for all types of summarization systems. Fourth, we provide an overview of various works from the literature and provide a comparative table of their obtained results. Finally, we describe the most influential automatic evaluation methods that rely or not on human references.

- *Chapter 3*. First, we introduce the automatic evaluation domain with a brief reminder of works from the literature. We focus on SERA, a method that assesses the quality of automatically generated summaries from the biomedical domain by comparing them to a set of reference summaries. This method is the basis of our contribution. Second, we present wikiSERA, our improved version of SERA, and an open-source system for evaluating summaries from the general domain. We explain how wikiSERA improves the query reformulation strategy with a Part-Of-Speech analysis of many corpora from different domains. Finally, we compare our approach with many works from literature and provide a comprehensive discussion of obtained results and some ablation studies. Extensive experiments related to this chapter are presented in Appendix A.

- *Chapter 4*. First, we introduce the automatic summarization domain while focusing on the limitations of current ATS systems. Second, we present our main contribution ($HazPi$). It is based on increasing the number of input decoders for faster text processing and better memory usage. Third, we present our second contribution concerning the second stage of training in which we encourage a relatively fast and progressive training of output summaries. Finally, we assess the performance of our approach with and without pre-training on a large dataset. We use as metrics the ROUGE approach, based on lexical overlaps, and both SERA and wikiSERA that are focused on the semantic content of summaries.

- *Chapter 5* - We finally discuss all chapters of the thesis briefly while presenting the main conclusions retained from this research. We finish the manuscript with an open window for future research in automatic text summarization and summary evaluation.

We first present our contribution in evaluation because we use it later to assess the quality of summaries generated by our proposed summarization approach.

Chapter 2

State of the art

2.1 Introduction

Automatic Text Summarization (ATS) is currently an active research area in Natural Language Processing (NLP). ATS task consists in capturing the most important information from a source text using an automatic system and reproducing it in the form of a shorter text. There are many ways to categorize Automatic Text Summarization systems:

1. Summarization can be either extractive or abstractive. Extractive approaches select the most relevant sentences from a source text and concatenate them to get a summary. The abstractive ones paraphrase the source text.

2. Summarization can be either mono-document or multi-document. Mono - document-based approaches summarize one text at a time, while multi-document-based ones summarize the content of multiple documents in one short paragraph.

3. Summarization can be either generic or query-based. Generic summaries aggregate information from the whole document, while query-based ones answer a specific question related to the document.

Over the years, researchers have developed several techniques to get automatic summaries. These techniques tackled complex problems, such as coherence and repetitions. The first approaches worked on extractive summarization, assuming that the most important words are repeated most frequently (Luhn, 1958, Sparck, 1972). The most important such methods include probabilistic models like, for instance, Probabilistic Context-Free Grammars (PCFG) (Rahman et al., 2001, Knight and Marcu, 2002), Markov Models, and Hidden Markov Models (HMM) (Chen and Withgott, 1992, Jing and McKeown, 1999, Conroy and O'leary, 2001).

Automatic summarization evolved later, and extractive based methods relied on machine learning to tackle ATS as a classification problem, where some techniques were used, such as Naive Bayes (Thu, 2014, Ramanujam and Kaliappan, 2016), Clustering (ShivaKumar and Soumya, 2015), and Support Vector Machine (SVM) (Schilder and Kondadadi, 2008, Begun et al., 2009). However, work was still needed to improve the automatic generation of summaries, especially with the new challenges that arose along with the emergence of neural networks.

Deep Neural Networks (DNNs) made it possible to generate abstractive summaries with the use of sequence-to-sequence models, such as Recurrent Neural Networks (RNNs) of type Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997), and Gated Recurrent Unit (GRU) (Cho et al., 2014). A typical sequence-to-sequence model can be seen as an encoding-decoding mapping from an input to an output sequence (Shaikh, 2018).

Researchers worked on improving recurrent neural networks for around two decades by introducing novel learning-rate scheduling functions, attention models, beam search (explained in Subsection 2.4.6.4), and modifications of the original neural networks (Cheng and Lapata, 2016, Zhenpeng, 2016, See et al., 2017, Nallapati et al., 2017). Alternatively, Vaswani et al. (2017) introduced Transformers, a novel neural network based on the attention mechanism.

Transformers outperformed state-of-the-art approaches released before 2017 and inspired several language models usable for automatic summarization, such as BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019), MASS (Song et al., 2019), UniML (Dong et al., 2019), and T5 (Raffel et al., 2020).

Meantime, researchers tackled challenges related to DNNs such as long input sequences, repetitions, and coherence of generated summaries. However, the lack of high-quality datasets limited the development of the learning methods. For this specific reason, long-text datasets were recently introduced, such as scientific articles (Cohan et al., 2018), newswires (Fabbri et al., 2019, Hermann et al., 2015, Nallapati et al., 2016), and medical corpora (Dernoncourt and Lee, 2017).

When summaries are generated automatically, their quality needs to be assessed. The evaluation can be done automatically or by humans. Manual evaluation is time and money expensive. Hence, researchers have developed two types of automatic evaluation systems: (1) those where human references are mandatory, and (2) those where the evaluation is fully automatic and does not rely on any human intervention.

This chapter is organized as follows: First, we explain the difference between automatic text summarization categories. Second, we describe datasets and corpora used whether in summarization or evaluation domains. Third, we present the most important works of state of the art in summarization and provide a quantitative comparison between some of them. Finally, we present the most influential works in automatic summary evaluation.

2.2 Automatic Summarization

It has been more than fifty years since the first research efforts were made in automatic text summarization. Since then, the amount of data has increased dramatically and so did the need for concise and widely available summaries (Kumar et al., 2016). In the following subsections, we expose different automatic summarization methods.

2.2.1 Extractive vs. Abstractive Summarization

We can distinguish between two families of summarization methods: extractive and abstractive. Extractive summarization is to "crop out and stitch together portions of the text to produce a condensed version of a text" (Rush et al., 2015). The summary is created by identifying and subsequently concatenating the most salient text units in a document (Cheng and Lapata, 2016). The pioneering work in these summaries was done by Luhn (1958). He used statistical information derived from word frequency and distribution to compute a relative measure of significance, first for individual words and later for sentences. Another important automatic text summarization system was done by Edmundson (1969). He used three methods for determining sentence weights: cue method, title method, and location method. Alternatively, Kupiec et al. (1995) extracted sentences based on many weightening heuristics. Extractive automatic summarization methods were used for decades. However, they often lead to problems in the overall coherence of the summary.

Abstractive summarization consists of generating a summary using novel words to explain the main idea of an article (Nallapati et al., 2016). To summarize is not only to extract chunks of text from original documents. It also refers to paraphrasing, generalizing, and incorporating new words. The biggest challenge for abstractive summarization is the text representation problem (Lin, 2009). For this reason, this type of automatic summarization has evolved from cognitive psycho linguistics and symbolic artificial intelligence to the use of neural networks and sequence models.

Abstractive summarization could have more similarities to the human summarization process than the extractive one (Chauhan, 2018).

2.2.2 Mono- vs. Multi-document Summarization

Automatic summarization techniques can be applied to one or more documents. The case of a single document was one of the first to emerge. Mono-document summarization relies on features like term frequency, sentence position, and stigma words. The multi-document case is more complicated to handle than the mono-document one because problems may arise in the summary or redundant information's coherence. However, this case has become more relevant given the growing amount of information and the need to summarize multiple documents in many domains: medical texts (Sarkar, 2009), news documents (Kumar and Salim, 2012), financial investments (Cardinaels et al., 2018), and conversations to improve the quality of the company's products and services (Tamura et al., 2011).

2.2.3 Generic vs. query-based Summarization

Text summarization can be generic or query-based. Generic summarization provides a summary of all the information contained in a document. Query-based summarization recovers partial information from a document based on a specific information need, like in a search engine where the answer to the question is presented with a predefined number of words (Vanetik and Litvak, 2017).

Before presenting state-of-the-art approaches, we present in the next section the most popular datasets and corpora used in automatic summarization and automatic evaluation of summaries.

2.3 Datasets for Automatic Summarization

Over the years, many datasets were built in Natural Language Processing to work on automatic summarization. One of the most important events is DUC (Document Understanding Conferences) (NIST, 2014). DUC was an international competition where the research community proposed novel methods to tackle NLP challenges, such as evaluating automatic summaries. These methods take into account reference summaries written by humans. This competition was held from 2001 to 2007, where each year, research groups used different corpora. In the next paragraph, we will describe the editions of DUC.

DUC01 contains 147 document-summary pairs where the summaries are designed for generic single-document extraction. DUC02 contains 567 document-summary pairs. The summaries are also for generic single-document extraction. DUC 2003 uses a dataset for automatic summarization that consists of 500 news articles from the New York Times and Associated Press Wire services. Each summary has four corresponding human reference summaries, consisting of 624 document-summary pairs. DUC 2004 has in the automatic summarization task 500 news pairs of articles and summaries from the New York Times (NYT) (Sandhaus, 2008) and the Associated Press Wire services. Each summary is associated with four human reference summaries. DUC 2006 contains 50 topics. Each topic is composed of 25 relevant documents from the AQUAINT corpus (Consortium, 2008). Documents belong to the news field and are mainly taken from the Associated Press, New York Times (1998-2000), and Xinhua News Agency (1996-2000). DUC 2007 (Over et al., 2007) is a dataset that aims to tackle two tasks. Having 25 documents in each of the 45 topics about news (Associated Press, New York Times (1998-2000), and Xinhua News Agency (1996-2000)), the main task is about question-answering based summarization, and the second one is about multi-documents short summary generation.

Another well-known dataset comes from TREC (Text REtrieval Conference) (NIST, 2020). TREC is a dataset for question classification. There are two TREC versions: with six classes (TREC-6) and fifty classes (TREC-50). Both of them have 5,452 training examples and 500 test examples.

The first attempt to get abstractive summaries in a sentence level was developed by Rush et al. (2015) using Gigaword (Napoles et al., 2012) dataset, which contains around 9.5 million news articles and four billion words. The articles are collected from seven sources: Agence France-Presse, Associated Press Worldstream, Central News Agency of Taiwan, Los Angeles Times/Washington Post Newswire Service, Washington Post/Bloomberg Newswire Service, New York Times Newswire Service, and Xinhua News Agency. Independently, New York Times (NYT) (Sandhaus, 2008) contains news articles from January 1st, 1987, to June 19th, 2007, from the New York Times (NYT). NYT consists of over 1.8 million articles. However, the library's scientists wrote over 650,000 article summaries. According to Cohan et al. (2018), the average number of words in the documents is 530, and in the abstracts is 38. Moreover, Gigaword inspired GIGA-CM (Zhang et al., 2019b). It is a database built from the English Gigaword dataset. The training/validation split is taken from the

CNNDM dataset, it contains 6,626,842 documents and 2,854 million words.

The scientific community decided to get abstractive summaries at the whole text level rather than at the sentence level. For instance, Hermann et al. (2015), Nallapati et al. (2016) introduced CNN/DailyMail News dataset. It is built from online news articles. Following Nallapati et al. (2016), there are 287,226 training pairs, 13,368 validation pairs, and 11,490 test pairs. The average number of tokens in the articles is 781, and in the abstracts is 56.

When Transformer neural networks emerged, many datasets did as well. For instance, Raffel et al. (2020) introduced the T5 model and the C4 dataset. It is a colossal and cleaned version of Common Crawl's web crawl corpus. It consists of 350 million of web-pages (750GB). More details about this dataset can be found on the TensorFlow (2020) website. Zhang et al. (2020a) used HugeNews to pre-train their model. HugeNews is a news collection corpus that contains 1.5 billion articles (3.8TB) from 2013-2019. It includes datasets such as XSum (Narayan et al., 2018a) and CNN/Daily Mail (Hermann et al., 2015). The news articles were acquired from news websites and blogs. Fabbri et al. (2019) introduced Multi-News dataset. It is a large-scale news dataset for multi-document summarization. It contains 56,216 news articles and model summaries written by professional editors.

Other well-known datasets are BillSum (Kornilova and Eidelman, 2019), XSum (Narayan et al., 2018a), NEWSROOM (Grusky et al., 2018), and WikiSum (Liu et al., 2018). BillSum is a dataset collected from the Congressional Research Service (CRS). It consists of three parts: U.S. training bills, U.S. test bills, and California test bills. BillSum contains 22,218 U.S. Congressional bills. Each U.S. bill has a human-written summary from the Congressional Research Service (CRS). XSum (Extreme Summarization) contains 226,711 news article-summary pairs collected from British Broadcasting Corporation (BBC) between 2010 and 2017. A summary is a single sentence that answers the question "*What is this article about?*" written by the article's authors (Narayan et al., 2018a). NEWSROOM is a dataset that contains 1.3 million articles and human-written summaries. Authors and editors wrote summaries in NEWSROOM from 38 major news publications. The authors collected the dataset from search and social media metadata between 1998 and 2017. WikiSum is a dataset from Wikipedia whose objective is to generate articles. It contains 2,332,000 articles and is made from two subsets of documents. The first one is taken from cited sources, and the second one is taken from web search results.

Over the years, researchers have been interested in the scientific domain whose characteristics are different from that of the general domain. For instance, scientific texts are longer than news ones and contain specialized terms and keywords. Examples of such medical datasets include Ziff–Davis and PubMed 200k RTC (Dernoncourt and Lee, 2017). Ziff–Davis corpus is a collection of newspaper articles announcing computer products. Some of the articles in the corpus are paired with human-written abstracts. PubMed 200k is a dataset for classifying sentences in medical abstracts. A label is assigned to each sentence's of an abstract, depending on the sentence role: background, objective, method, results, or conclusion. The authors collected the abstracts from the PubMed website in 2016, which belong to the RTC (Randomized Controlled Trials). The dataset contains 195,654 abstracts.

A couple of years later, Cohan et al. (2018) introduced two datasets. The first one is arXiv, a dataset of long scientific papers collected from the arXiv website. It consists of 215K documents, where the average number of words in the documents is 4,938, and the average number of words in the abstracts is 220. The second is the PubMed dataset. It contains long documents with discourse information. The abstracts of the articles are used as gold summaries. The dataset consists of 133K documents, having an average of 3,016 words per document and 203 words per abstract. Another scientific dataset is Multi-XScience (Lu et al., 2019), a multi-document abstractive summarization dataset inspired by XSum corpus. It consists of 40,528 scientific articles. The dataset is built from Microsoft Academic Graph (MAG) (Sinha et al., 2005) and arXiv (arXiv, 2021).

Besides the scientific datasets, researchers introduced corpora about patents, kitchen recipes, opinions from users, and email messages such as BIGPATENT (Sharma et al., 2019), Reddit TIFU (Kim et al., 2019), WikiHow (Koupaee and Wang, 2018), and AESLC (Zhang and Tetreault, 2019). BIGPATENT consists of 1.3 million U.S. patent documents with human-written abstractive summaries. The patent description is the input, and the patent's abstract is the gold summary. The documents are collected from Google Patents Public Datasets. Reddit TIFU dataset consists of 122,933 posts published from January 2013 to March 2018 in the online discussion forum "Reddit". Posts are source texts and have a corresponding long or short summary written by the same user. WikiHow is an abstractive dataset that contains 204,004 articles and summaries written by humans. The authors collected the dataset from the online WikiHow knowledge base. The articles describe a procedural task about various topics from 20 categories. AESLC is the annotated version of the Enron dataset (Klimt and Yang, 2004). It is a collection of email messages from employees in the Enron Corporation. AESLC uses the email body as the source text and the email subject line as the gold summary. Only emails with at least three sentences and 25 words in the email body are considered. Therefore, the dataset contains 18,302 pairs, where humans evaluate 500 samples.

Once we reviewed the most popular corpora by the scientific community, we present the automatic text summarization approaches in the state of the art in the following section.

2.4 Methods for Automatic Summarization

2.4.1 Frequency Based Approaches

Luhn (1958) was one of the first to work on automatic summarization. He assumed that the most important words are repeated most frequently in a text and can be used to build the summary.

2.4.1.1 Word Probability

This technique consists of counting the number of times each word appears in the document and then computing its probability as follows:

$$f(w) = \frac{n(w)}{N} \tag{2.1}$$

where:

- $n(w)$ is the frequency of the word w
- N is the total number of words in the document

A weight can also be attributed to a sentence S_j as following:

$$weight(S) = \frac{\sum_{w \in S} f(w)}{|\{w|w \in S\}|} \tag{2.2}$$

2.4.1.2 TF-IDF

This approach is a product of two terms $TF \times IDF$

- TF (Term-Frequency) is the number of times that a term appears in the document. According to Kumar et al. (2016), TF is defined as following:

$$TF(t,d) = \frac{f_{t,d}}{\sum\limits_{t' \in d} f_{t',d}} \tag{2.3}$$

  where $f_{t,d}$ is the frequency of the term t in document d.

- IDF (*Inverse Document Frequency*) was proposed for the first time in 1972 by Sparck (1972). It attenuates the weight of terms that appear very frequently in documents and increases the weight of terms that occur rarely. IDF is defined as follows:

$$IDF(t,D) = log\frac{|D|}{|\{d \in D : t \in d\}|} \tag{2.4}$$

  where $|D|$ is the total number of documents and $|\{d \in D : t \in d\}|$ is the number of documents that contain the term t.

Therefore:

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,D) \tag{2.5}$$

The first works in summarization were based on $TF - IDF$ such as Luhn (1958). Over time, $TF - IDF$ was included in other complex techniques. For instance, Erkan and Radev (2004) worked on getting extractive summaries. They compute the centrality of each sentence in the text. We can compute the sentence centrality in terms of the centrality of the words contained by each sentence. The words' centrality is related to the centroid set of documents. A threshold of $TD - IDF$ determines the salience of each word.

2.4.2 Feature Based Approaches

One way to determine a sentence's relevance is to identify features that reflect the importance of the sentence. According to Kumar et al. (2016), the following features are essential to determine the most relevant sentences: (1) Title/Headline Word, (2) Sentence Position, (3) Sentence Length, (4) Term Weight, and (5) Proper Noun.

2.4.3 Probabilistic Models

A probabilistic language model defines a probability distribution on the set of characters or strings based on a corpus analysis (text collection). Each element has an associated probability, and these probabilities are learned from a corpus. We present three probabilistic models: Context-Free Grammars, Markov Models, and N-gram models.

2.4.3.1 Probabilistic Context Free Grammars

Probabilistic Context-Free Grammars (PCFG) are a probabilistic model of syntax for tree structures. A context-free grammar consists of (Manning and Schutze, 1999):

- A set of terminals w^k, where $k = 1, ..., V$ and V is the vocabulary size
- A set of non-terminals N^i, where $i = 1, ..., n$, and n the the number of non-terminals
- A designed start symbol N^1
- A set of rules $\{N^i \rightarrow \zeta^j\}$, where ζ^j is a sequence of terminals and non-terminals
- A corresponding set of probabilities on rules such that $V_i \sum P(N^i \rightarrow \zeta^j) = 1$

The sentence to parse is represented as a sequence of words $w_1, .., w_m$ and w_{ab} is a subsequence $w_a...w_b$. Using the rules of grammar, we can derive sentences and represent this derivation through a tree (Figure 2.1).

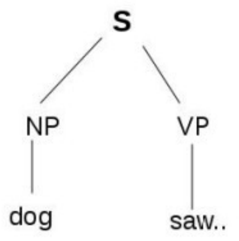


Figure 2.1: Derivation tree built from a Context Free Grammar

The probability of a sentence (according to a grammar G) is computed by Equation 2.6 (Manning and Schutze, 1999):

$$P(w_1...w_m) = \sum_t P(w_1...w_m, t) \tag{2.6}$$

where $w_1...w_m$ is a sentence to be parsed and t is a parse tree of the sentence.

We can use a grammar to generate text or analyze (parse) it. PCFGs have many advantages, such as their effectiveness for grammar induction and their ability to avoid some problems such as grammatical mistakes and disfluencies. However, they suffer from some disadvantages like, for instance, the lack of sensitivity to lexical information and structural frequencies.

Rahman et al. (2001) worked on automatic summarization of web pages. They used PCFG to define syntactic structures, analyze and understand the content, and determine its importance. Alternatively, Knight and Marcu (2002) focused on sentence compression. They developed a probabilistic noisy-channel model that used PCFG to assign probabilities to a tree.

2.4.3.2 Markov Model

Markov Model is a stochastic model in which an unknown (hidden) future value is predicted in a Markov sequence (chain). The value to predict depends only on the immediate previous value. X is a Markov chain if $X = (X_1, ..., X_T)$ is a sequence of random variables that take values in a finite set $S = \{s_1, ..., s_N\}$ and fulfill the following properties describe in the Equations 2.7 and 2.8 (Manning and Schutze, 1999):

- *Limited Horizon*

$$P(X_{t+1} = s_k | X_1, ..., X_t) = P(X_{t+1} = s_k | X_t) \tag{2.7}$$

- *Time invariant (stationary)*

$$P(X_{t+1} = s_k | X_1, ..., X_t) = P(X_2 = s_k | X_1) \tag{2.8}$$

where P is the probability.

Example: Given the sentence: "*The dog is big*". The time invariant is: $P(dog|The)$, $P(is|dog)$ and $P(big|is)$, because the order of the words does not change.

We can represent Markov Models as probability equations or as state diagrams.

2.4.3.3 Hidden Markov Models

The original goal of Hidden Markov Models (HMMs) was to model the letter sequences in Alexander Pushkin's poetry in 1913. The model's state sequence is unknown in an HMM, but only some of its probabilistic functions.

Following the notation of Manning and Schutze (1999), we consider the general form of an HMM as

$$(S, K, II, A, B)$$

where S and K are the set of states and the output alphabet, respectively. II, A, and B are the probabilities of the initial state ($II = \{\pi_i\}, i \in S$), state transitions ($A = \{a_{ij}\}, i, j \in S$) and symbol emissions probabilities ($B = \{b_{ijk}, i, j \in S, k \in K\}$), respectively.

To find the best state sequence, it is possible to use the *Viterbi* algorithm (Equation 2.9) (Manning and Schutze, 1999), which can help in computing the most likely state sequence.

$$\delta_j(t) = max_{x_1 \ldots x_{t-1}} P(X_1 \ldots X_{t-1}, o_1, ..., o_{t-1}, x_t = j|\mu) \tag{2.9}$$

where:

$O = o_1, ..., o_T$ is the observation sequence.

HMMs are a robust tool when combined with efficient algorithms such as Expectation - Maximization (EM). Besides, they can be used to generate parameters for linear interpolation of n-gram models.

Chen and Withgott (1992) applied Hidden Markov Models on speech summarization. Their method is based on identifying emphasized speech and then using proximity measures to select summarizing fragments. Alternatively, Jing and McKeown (1999) proposed an algorithm based on HMM that decomposes human-written summary sentences. The goal is to determine the relations between sentences in human-written summaries and sentences in the original text. Also, Conroy and O'leary (2001) proposed a method for text summarization that considers three features: (1) the position of the sentence in the document (using Hidden Markov Model), (2) the number of terms in the sentence, (3) the probability of the terms. The method aims to compute the overall sentence probability and decide if it belongs to the summary.

2.4.3.4 N-gram models

N-gram models are Markov Models. As mentioned earlier, Markov models are used to predict a future value in a sequence. We can predict the next word in a sequence using Equation 2.10 (Russell and Norving, 2010):

$$P(w_1, ..., w_n) = P(w_1)P(w_2|w_1) \ldots P(w_n|w_1, ..., w_{n-1}) \tag{2.10}$$

These probabilities are learned from a corpus.

Unigram Model As its name indicates, we can deduce each word's probability independently (Equation 2.11).

$$P(w_1...w_n) = \prod_i P(w_i) \tag{2.11}$$

$$\text{with } P(w_i) = \tfrac{N(w_i)}{N}$$

where $N(w_i)$ is the number of times the word w_i appears in the corpus, and N is the total number of words (including repetitions).

Bigram Model We can compute the probability of a word knowing the previous one but independently of the other words (Equation 2.12).

$$P(w_1....w_n) = P(w_1) \prod_i P(w_{i+1}|w_i) \tag{2.12}$$

$$\text{with } P(w_j|w_i) = \tfrac{N(w_i w_j)}{N(w_i)}$$

where $N(w_i w_j)$ is the number of occurrences of the bigram (consecutive words $w_i w_j$) in the corpus, and $N(w_i)$ is the frequency of the word w_i in the corpus.

Trigram Model In a Trigram model, we can get the probability of a trigram (three consecutive words) by computing the probability of a word knowing the two immediate preceding ones (Equation 2.13).

$$P(w_1...w_n) = P(w_1)P(w_2|w_1) \prod_i P(w_{i+2}|w_{i+1}, w_i) \tag{2.13}$$

N-gram Model The generalization of the previous models is formed by n consecutive words in the corpus. In these probabilistic models, except for the unigram model, lexical-contextual relationships are taken into account.

2.4.4 Smoothing in n-gram models

The main disadvantage of N-gram models is their disability to handle Out-Of-Vocabulary (OOV) terms. When a word does not belong to the training set vocabulary, the language model associated with it tends to have zero probability, causing the whole product's cancellation. The goal of smoothing techniques is to avoid zero probabilities produced by unseen n-grams.

2.4.4.1 Laplace's law

The most straightforward smoothing technique consists of adding 1 to the numerators of the individual probabilities and appropriately compensating the total sum by increasing the denominators (Manning and Schutze, 1999).

- **Smoothing Unigram Models**

  $$P(w) = \frac{N(w)+1}{N+V_1} \tag{2.14}$$

  where V_1 is the total number of words in the corpus.

- **Smoothing Bigram Models**

  $$P(w_j|w_i) = \frac{N(w_iw_j)+1}{N(w_i)+V_2} \tag{2.15}$$

  where V_2 is the total number of bigrams in the corpus.

- **Smoothing Trigram Models**

  To compute trigrams probability, we use a combination between bigram and unigram (Equation 2.16).

  $$P(w_3|w_1,w_2) = \lambda_3 P_3(w_3|w_1,w_2) + \lambda_2 P_2(w_3|w_2) + \lambda_1 P_1(w_3) \tag{2.16}$$

  where P_1, P_2, and P_3 are Unigram, Bigram and Trigram probabilities, respectively, and $\lambda_1 + \lambda_2 + \lambda_3 = 1$.

Villatoro-Tello et al. (2006) represent sentences by word sequences (n-grams). This approach improved the performance in automatic summarization. However, considering sentences as a set of n words helps only in extractive approaches.

2.4.5 Machine Learning Approaches

In Machine Learning (ML), extractive automatic summarization can be handled as a binary classification problem. Each sentence in the text is represented as a numerical vector before being fed into the model. For each sentence, we associate a zero-label (summary sentence) if the sentence belongs to the reference summary and a one-label (non-summary sentence) otherwise (Kumar et al., 2016).

2.4.5.1 Naive Bayes

Naive Bayes is a classification technique that constructs models by predicting conditional probabilities. Kupiec et al. (1995) are one of the first to apply this algorithm to the automatic summarization (Equation 2.17). Given a sentence s, its probability of being included in the summary is:

$$P(s \in S|F_1, F_2, ..., F_n) = \frac{\prod_{i=1}^{n} P(F_i|s \in S).P(s \in S)}{\prod_{i=1}^{n} P(F_i)} \tag{2.17}$$

where $F_1, F_2, ..., F_n$ are sentences for the classification and S is the summary to generate.

Ramanujam and Kaliappan (2016) extended the application of the Naive Bayes algorithm by combining it with the timestamp approach for the automatic summarization of multi-documents. Obtained summaries were better in terms of coherence.

2.4.5.2 Clustering

Clustering is a type of unsupervised learning method. It consists of splitting a set of objects into non-overlapping groups called clusters (Manning and Schutze, 1999) to put similar objects in the same group. Clustering requires similarity metrics, which are often computed at the word level in NLP.

One similarity technique is to use the whole distributional patterns of words to measure the degree of overlap in the neighborhood distributions of two words.

Following Manning and Schutze (1999), many clustering algorithms can be classified in two different ways: (1) hierarchical clustering vs. flat clustering and (2) hard clustering vs. soft clustering. In flat clustering, we set in advance the number of clusters. However, in hierarchical clustering, we do not pre-define the number of clusters. In hard clustering, each element belongs to one and only one cluster, which is not the case for soft clustering.

Aliguliyev (2009) worked on extractive summarization. They proposed a method based on sentence clustering. According to the content of the cluster, it identifies the most salient sentences. Similarly, ShivaKumar and Soumya (2015) worked on extractive summarization where authors generate the document clusters based on the similarity between the documents. Then, they pick sentences with the best scores from each cluster and add them to the summary.

2.4.5.3 Support Vector Machines

Support Vector Machines (SVMs) are supervised algorithms proposed by Kecman (2005). These models can be used to solve classification and regression problems. An SVM can predict the new sample's class from the previous training samples. An SVM is a model that separates data points into classes by a hyperplane called a support vector.

An ideal problem for an SVM consists of two classes, that can be separated by a straight line (see Equation 2.18 and Figure 2.2).

$$(x_1, y_1), (x_2, y_2), ..., (x_l, y_l), x \in \mathbb{R}^2, y \in \{+1, -1\} \tag{2.18}$$

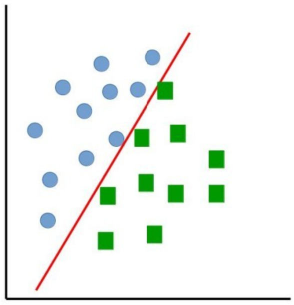


Figure 2.2: Linear splitting of a 2-dimensional set using a Support Vector Machine

When it comes to nonlinear classifiers, advanced algorithms can be used, such as Euclidean space and Hilbert space (also known as *maximum margin classifiers*), that can make an *optimal separation.*

Schilder and Kondadadi (2008) worked on query-based multi-document summarization using SVMs to rank all sentences in the topic cluster. The summary is then constructed by concatenating sentences with high scores. Similarly, Begun et al. (2009) worked on automatic text summarization using SVMs. To train their model, the authors extracted features from the text, such as the sentence's position, the centrality of the sentence, and the sentence's resemblance with the title.

2.4.6 Deep Learning approaches

In the following sections, we describe some ATS approaches based on deep learning. The latter refers to neural networks with several layers (dozens to hundreds).

2.4.6.1 Encoder-decoder models

Sutskever et al. (2014) introduced sequence-to-sequence models that aim to map input tokens to output tokens. The encoder-decoder model is a way of using Recurrent Neural Networks (RNNs, explained in Subsection 2.4.6.2) for sequence-to-sequence problems.

A sequence-to-sequence model has three components: an encoder, an intermediate (encoded) vector, and a decoder. We define each one of them as follows (Kostadino, 2019):

Encoder - is a stack of many recurrent neural networks that takes as input the text to summarize.

Encoded vector - (also called context vector) is the output of the encoder and the input of the decoder.

Decoder - is also a stack of many recurrent neural networks. The decoder receives the encoded vector and the gold standard and produces a summary.

In the case of translation and summarization, input and output sequences have possibly different lengths. For instance, when translating the sentence "I like it" from English to Spanish, "me gusta", the 3-tokens English phrase is the encoder's input, and the 2-tokens Spanish phrase is the decoder's output (see Figure 2.3).

The emergence of encoder-decoder models improved the state of the art in both translation and summarization, where there are two relevant sequences. The first one is the text to translate or summarize, and the second one is the gold standard (translated text or reference summary).

In the following subsection, we explain recurrent neural networks and their leading derivatives: LSTMs and GRUs.

2.4.6.2 Recurrent Neural Networks

Recurrent Neural Networks (RNNs, presented in Figure 2.4) are deep neural networks that take sequential steps to encode and decode an input token by token.

Unfortunately, RNNs cannot process a sequence parallelly. Besides, since the number of time steps in the RNN corresponds to the number of tokens in the sequence, the longest the sequence, the more the RNN takes time to encode it. Also, long sequences lead to information loss because of the vanishing gradient problem.

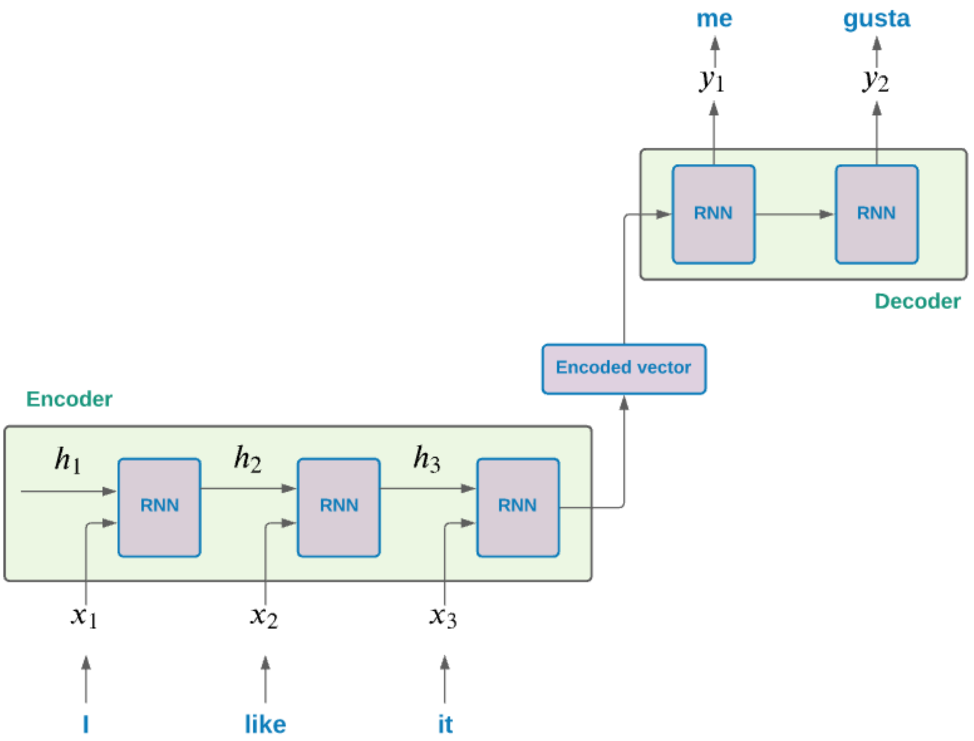


Figure 2.3: Sequence-to-sequence model

The vanishing gradient problem. Like all types of neural networks, we train a recurrent neural network with the help of a loss function $\mathcal{L}$ that is used to optimize the model's parameter values. A loss function quantifies the error between the output predicted by the neural network and the target (Equation 2.19).

$$\mathcal{L} = \sum_i \mathcal{L}_i(\hat{y}_t, y_t) \tag{2.19}$$

where:

$\mathcal{L}_i$ is the loss at time step i

$\hat{y}_t$ is the target (ground-truth)

y_t is the model's output

Once the loss is computed, we minimize it by back-propagating its gradient through the RNN layers and also through time. Hence, at each time step, we have to sum up all the previous gradients, as shown in Equation 2.20.

$$\frac{\partial L}{\partial W} = \sum_{i=0}^{T} \frac{\partial \mathcal{L}_i}{\partial W} \propto \sum_{i=0}^{T} \left(\prod_{i=k+1}^{y} \frac{\partial h_i}{\partial h_{i-1}} \right) \frac{\partial h_k}{\partial W} \tag{2.20}$$

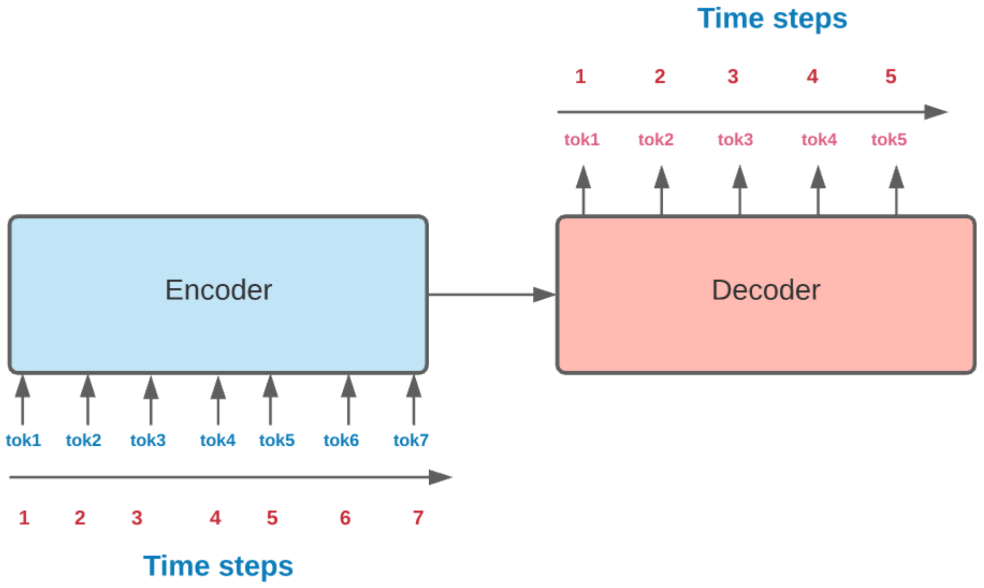


Figure 2.4: RNNs way to handle sequence inputs

where:

W is the weight matrix

T is the total number of time steps

h_k is the hidden state at time step k

In Equation 2.20, the contribution of a state at time step k to the gradient of the entire loss function $\mathcal{L}$, at time step $t = T$ is calculated.

The vanishing gradient problem occurs when the part of the equation highlighted in red tends to zero quickly. In such a case, it is challenging to learn long data sequences. Two types of RNNs were born to help tackle the vanishing gradient problem: LSTM and GRU.

LSTM (Long Short-Term Memory) networks were introduced by Hochreiter and Schmidhuber (1997). They are used for sequential tasks such as machine translation and language modeling. These recurrent neural networks can read long sequences compared to RNNs. Besides, unlike RNNs, LSTMs have more than one hidden state, which can help avoid the vanishing gradient problem.

In addition to work with long sequences, LSTMs have better control in memory management than RNNs. This is because an LSTM can select which information to store, update, delete, or forget through its components: cell state (memory), hidden state (used to calculate predictions), input gate, forget gate, and output gate. Figure 2.5 depicts LSTM components and how they are related.

As mentioned before, LSTM has three gates which are Sigma (σ) functions. A Sigma function takes values from 0 to 1. It represents how much information will flow. If the gate value is 0, the information will not flow, and if it is 1, the complete information will flow. The following list describes the functionality of each gate and its equation.

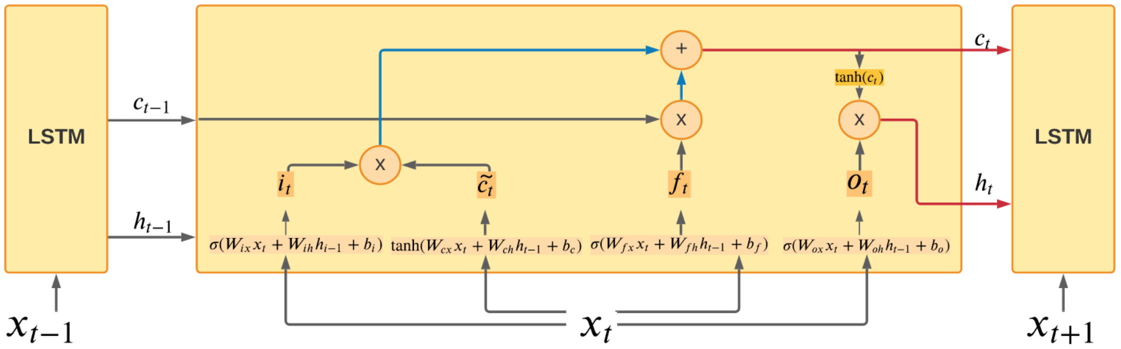

LSTM
c_{t-1}
h_{t-1}
x_{t-1}
+
X
X
X
tanh(c_t)
c_t
h_t
i_t
$\tilde{c}_t$
f_t
o_t
$\sigma(W_{ix}x_t + W_{ih}h_{i-1} + b_i)$
$\tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c)$
$\sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f)$
$\sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o)$
x_t
LSTM
x_{t+1}

Figure 2.5: LSTM Neural Network

- **Input gate** regulates how much information the current input will read into the cell state.

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{i-1} + b_i) \tag{2.21}$$

where:

t is the time step

i_t is the input gate in the t^{th} time step

W is the weight matrix

x is a training sample

h is the hidden state

b is the bias vector

- **Forget gate** regulates how much information of the previous cell state will pass into the current cell state.

$$\tilde{f}_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \tag{2.22}$$

- **Output gate** regulates how much information of the cell state will pass into the hidden state.

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \tag{2.23}$$

However, the main function of these gates is to update the current cell state and determine the final hidden state. The cell state is described in the Equation 2.24 while Equation 2.26 describes the final hidden state.

$$c_t = f_t c_{t-1} + i_t \tilde{c_t} \tag{2.24}$$

where $\tilde{c_t}$ is the candidate value (Equation 2.25):

$$\tilde{c_t} = \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \tag{2.25}$$

$$h_t = o_t \tanh(c_t) \tag{2.26}$$

The final output (prediction) is computed using Equation 2.27.

$$y_i = softmax(W_s h_t + b_s) \tag{2.27}$$

Over the years, researchers have improved LSTM neural networks to get better predictions. These improvements include bidirectional LSTMs, Beam Search (Graves, 2012, Boulanger-Lewandowski et al., 2013, Sutskever et al., 2014), and the use of word vectors (Mikolov et al., 2013, Peters et al., 2018).

GRU (Gated Recurrent Unit) was introduced by Cho et al. (2014). GRU is a simplification of LSTM. In the following, we describe the differences between GRUs and LSTMs.

- GRU combines the cell state and the final hidden state of an LSTM into a single hidden state.

- GRUs introduced a reset gate, which is a Sigma function. If the GRU value is 1, all the previous state information is used to compute the current state. If the GRU value is 0, all data from the previous state is ignored.

- GRU introduced an update gate, which is a combination of the input gate and the forget gate. If the update gate value is 0, the current state uses all the information from the previous state, and nothing from the input is read into the current state. Inversely, if the update gate value is 1, all the current input is read, and the previous state's information is not considered.

Lots of works have used LSTMs and GRUs for automatic summarization. LSTM was used by Cheng and Lapata (2016) and Zhenpeng (2016). On the one hand, Cheng and Lapata (2016) used LSTM for the extractive summarization of single documents. Their decoder chooses output symbols from the document of interest rather than the entire vocabulary. On the other hand, Zhenpeng (2016) applied a hierarchical LSTM model to build the sentence representations in abstractive and long summaries.

GRUs were used by Nallapati et al. (2017). The authors proposed SummaRuNNer (simple recurrent network-based sequence classifier) for extractive summarization and used two-layer bi-directional GRU as the basic building block of their sequence classifier.

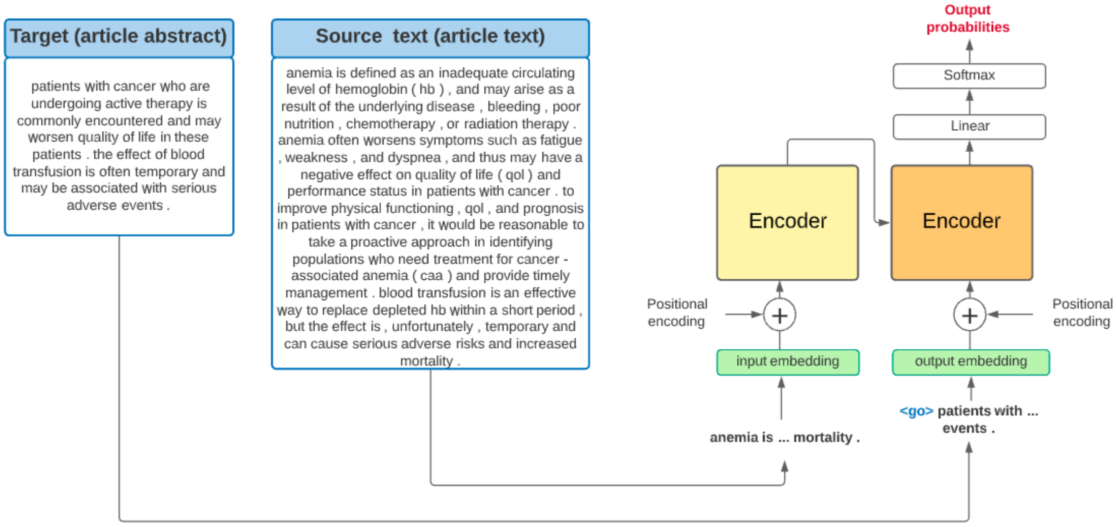
Despite the evolution of LSTM and GRU neural networks, there are still some drawbacks. For instance, information loss for long sequences and the processing time of a sequence depends on its length since these kinds of neural networks read sequences token by token.

In LSTM and GRU neural networks, it is not possible to parallelize sequence reading. For this reason, a new type of neural network was born to tackle these drawbacks: Transformers (Vaswani et al., 2017).

2.4.6.3 Transformers

Transformers were introduced by Vaswani et al. (2017) to tackle some problems that RNNs suffer from, such as loss of information with long sequences and the vanishing gradient. A Transformer is a deep neural network based on an encoder and a decoder (Section 2.4.6.1). Figure 2.6 shows a simplified representation of a Transformer, while Figure 2.7 details the architecture of the encoder and the decoder.

The encoder is a set of six layers, where each layer contains two sub-layers: a multi-head attention layer and a feed-forward network. The decoder also has six layers but is different from the encoder in two aspects. First, it has an additional multi-head attention sub-layer, and second, the self-attention sub-layer is modified to avoid attending subsequent positions. Each sub-layer of the encoder and the decoder is followed by a residual connection and a normalization layer. Each layer


Target (article abstract)

patients with cancer who are undergoing active therapy is commonly encountered and may worsen quality of life in these patients . the effect of blood transfusion is often temporary and may be associated with serious adverse events .

Source text (article text)

anemia is defined as an inadequate circulating level of hemoglobin (hb) , and may arise as a result of the underlying disease , bleeding , poor nutrition , chemotherapy , or radiation therapy . anemia often worsens symptoms such as fatigue , weakness , and dyspnea , and thus may have a negative effect on quality of life (qol) and performance status in patients with cancer . to improve physical functioning , qol , and prognosis in patients with cancer , it would be reasonable to take a proactive approach in identifying populations who need treatment for cancer - associated anemia (caa) and provide timely management . blood transfusion is an effective way to replace depleted hb within a short period , but the effect is , unfortunately , temporary and can cause serious adverse risks and increased mortality .

Figure 2.6: Transformers General Architecture

of the encoder and the decoder has a fully connected feed-forward network. All the sub-layers in the model and the embedding layers produce outputs of the same dimension (called d_{model}).

The first sub-layer of both an encoder and a decoder is the multi-head attention mechanism (a set of self-attentions). We first start by explaining how the sequence input flows through a self-attention mechanism, and later we explain how these self-attention heads are concatenated together.

Given an input sequence, each token from this sequence is converted to a fixed-size vector using an embedding algorithm. In practice, we concatenate embedding vectors to a matrix of size $(batch\ size \times d_{model})$.

Unlike recurrent neural networks, Transformers do not contain time steps to retain tokens order in the input sequence. Instead, they rely on positional encoding embeddings of dimension d_{model}. The latter vectors are summed with input embeddings at the bottom of the encoder and the decoder stacks (Figure 2.8). We can compute the positional encoding with Equation 2.28 or Equation 2.29.

$$PE_{pos,2i} = sin(pos/10000^{2i/d_{model}}) \tag{2.28}$$

$$PE_{pos,2i+1} = cos(pos/10000^{2i/d_{model}}) \tag{2.29}$$

where pos is the position of the token in the sequence and i is the positional encoding dimension.

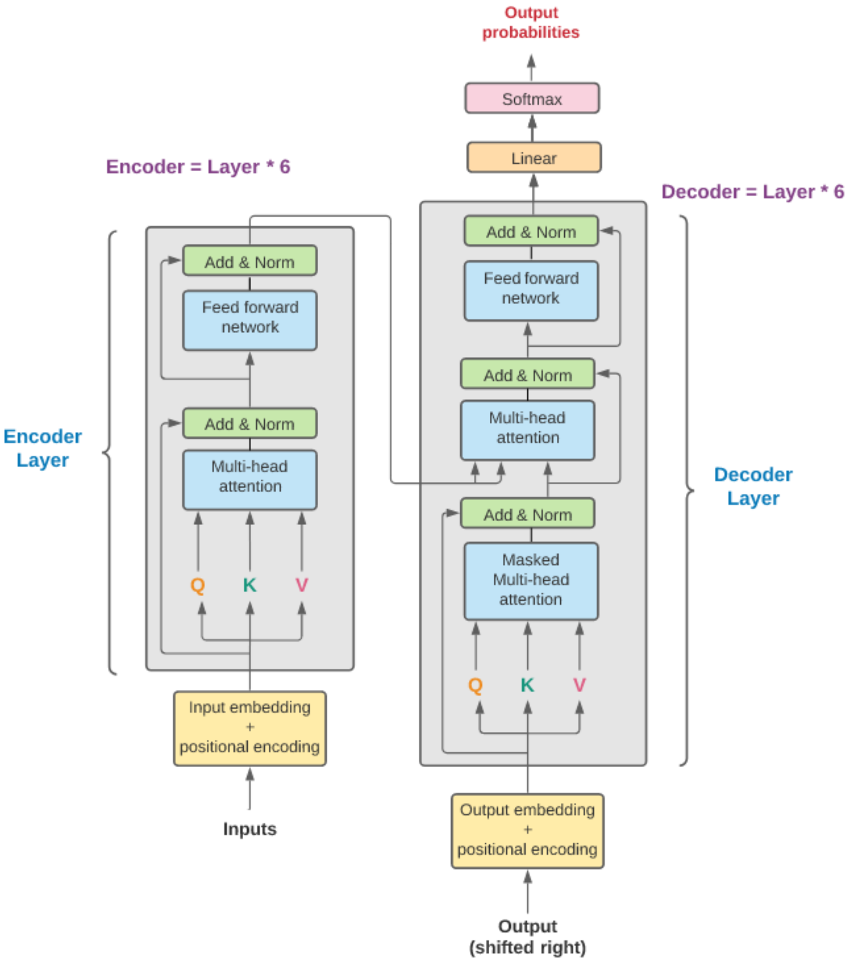


Figure 2.7: Transformer architecture (Vaswani et al., 2017)

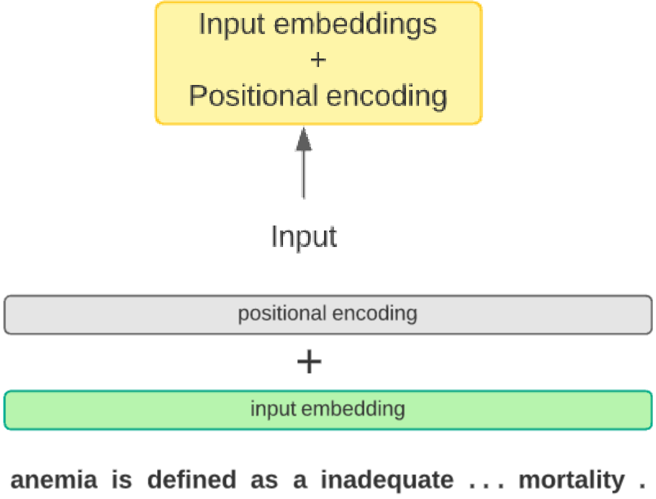

Input embeddings + Positional encoding
Input
positional encoding
+
input embedding
anemia is defined as a inadequate . . . mortality .

Figure 2.8: Positional encoding in Transformers

Once the input of the Transformer is ready (embedding + positional encoding), the self-attention is computed as in Figure 2.9.

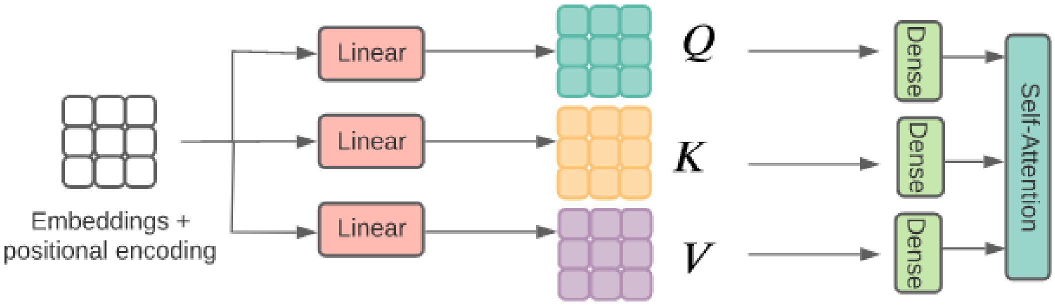


Figure 2.9: From input to self-attention

As shown in the figure above, self-attention is a model that integrates three fully connected layers by which the input flows in order to get Q (queries), K (keys), and V (values) matrices.

Equation 2.30 shows how self-attention is computed.

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2.30}$$

where Q and K contain vector representations of each word in the sequence, and V contains values from each word in the sequence.

Therefore, attention weights come from a dot product between queries (Q) and keys (K) matrices. A softmax function is needed to convert these weights into probabilities. The attention weights indicate how much each key is similar to each query.

As mentioned above, multi-head attention is a set of self-attentions (heads) concatenated as shown in Figure 2.10 and Equation 2.32. These layers simulate the recurrence effect with attention.

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)\ W^O \tag{2.31}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{2.32}$$

where:

- h is the number of heads
- $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$, and $W_O \in \mathbb{R}^{h_v \times d_{model} \times d_v}$

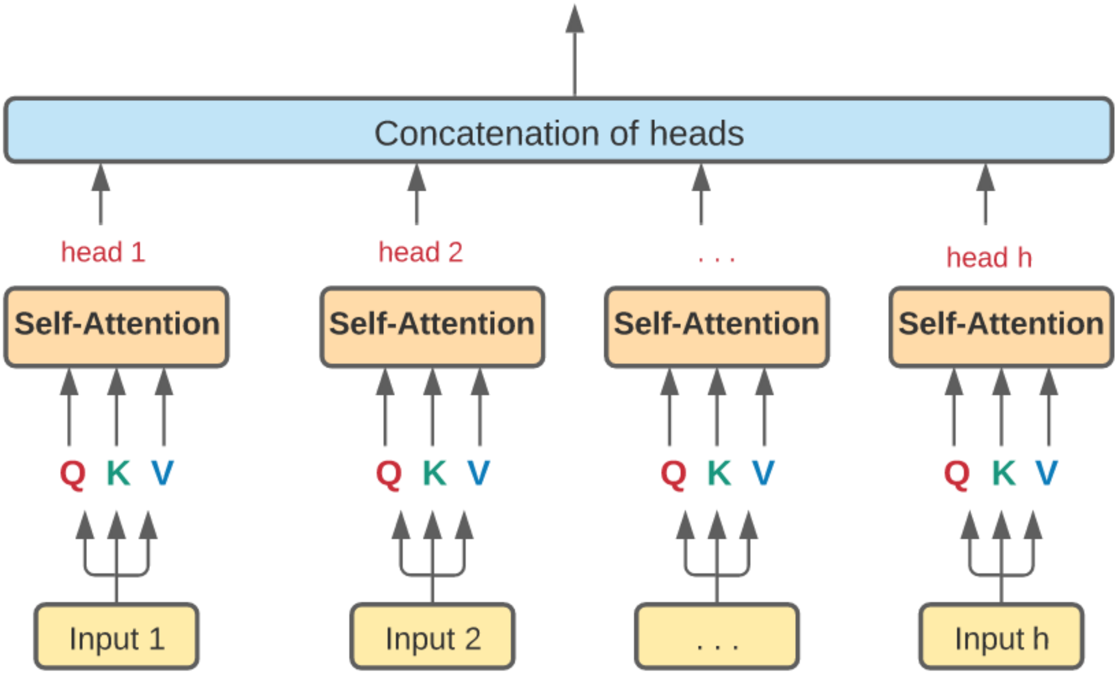


Figure 2.10: Multi-head attention

- $d_k = d_v = d_{model}/h$

There are three different ways to handle multi-head attention in Transformers: (1) at the encoding level only, (2) at the decoding level only, or (3) at both the encoding and the decoding levels.

The second transformer sub-layer of each layer in both the encoder and the decoder is a fully connected feed-forward network. It is applied to each position separately and identically.

Transformer neural network has improved state of the art in various tasks, such as summarization. Its success has been increasing with the use of pre-trained models such as BERT (Devlin et al., 2019), T5 (Raffel et al., 2020), and BART (Lewis et al., 2019). Inspired by BERT, Zhang et al. (2019b) introduced HIBERT (HIerarchical Bidirectional Encoder Representations from Transformer), an automatic system for abstractive summarization, where the authors introduced some noise in the text, and the model is trained to rebuild the source text.

In the following, we present the most influential Transformer-based approaches that are adaptable for automatic text summarization.

BERT Inspired by Transformers, BERT (Vaswani et al., 2017), (Bidirectional Encoder Representations from Transformers) was introduced by (Devlin et al., 2019). It is a bidirectional model that is based on an encoder only.

In BERT, it is possible to use pre-trained tasks such as multi-mask language modeling and sentence prediction. Multi-mask language consists of masking some words in the sentences. Later, the neural network attempts to predict the masked words. Figure 2.11 describes both of them.

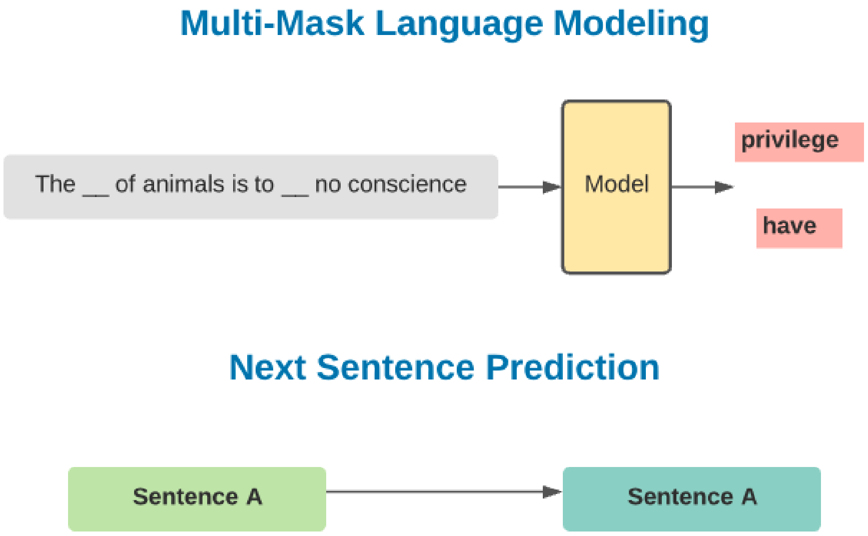

Multi-Mask Language Modeling
The __ of animals is to __ no conscience
Model
privilege
have
Next Sentence Prediction
Sentence A
Sentence A

Figure 2.11: Multi-Mask Language Modeling and Sentence Prediction in BERT

- BERT's input is a sum of position, segment, and token embeddings. There are two special tokens used in BERT: $[CLS]$ and $[SEP]$. These tokens indicate the beginning and the end of the sentence, respectively. Note that $[SEP]$ serves as a sentence-separator.

- There are two main goals in BERT. The first one is Multi-Mask Language Modeling (LM). The second one is the Next Sentence Prediction. For Multi-Mask LM, the authors used Cross-Entropy Loss to predict masked words and Binary Loss for Next Sentence Prediction.

- We can fine-tune BERT to transfer knowledge between different tasks. Once BERT is pre-trained, we can use it, for example, in sentiment analysis, Multi-Genre Natural Language Inference (MNLI), Named-entity recognition (NER), question-answering, and summarization.

T5 (Raffel et al., 2020) is an encoder-decoder architecture inspired by Vaswani et al. (2017). T5 is designed for multi-task learning (learning many tasks at once) in a bidirectional context. Unlike Transformers, T5 consists of 12 block layers in the encoder/decoder, resulting in 220 million parameters.

When T5 is pre-trained on a multi-tasking mix, we can use it for classification, question answering, machine translation, summarization, and sentiment analysis. Figure 2.12 shows some of T5 applications.

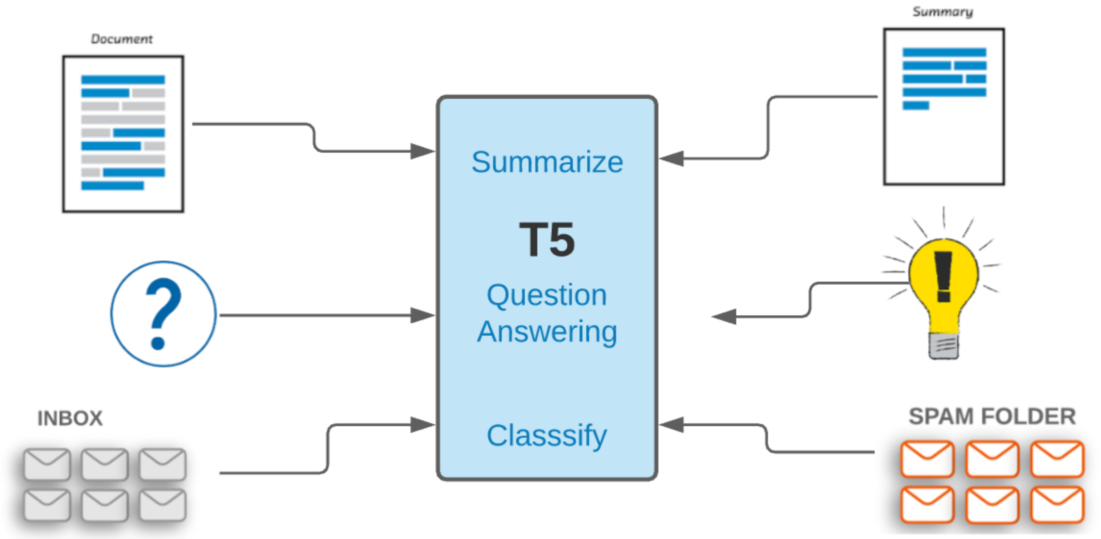
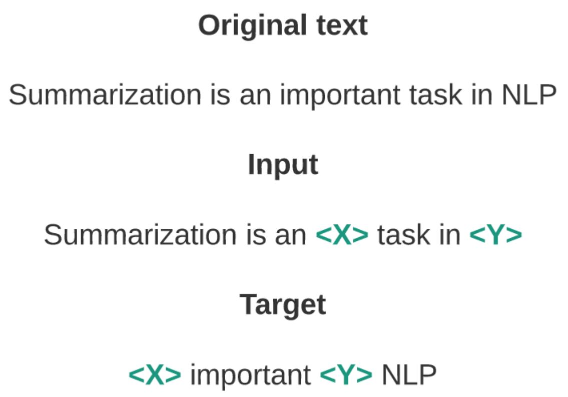


Figure 2.12: T5 applications

To avoid overfitting, T5 masks some tokens from the original text using ordered special tokens: <A>, <B>, <C>,..., etc. We provide an example in Figure 2.13.



Original text

Summarization is an important task in NLP

Input

Summarization is an <X> task in <Y>

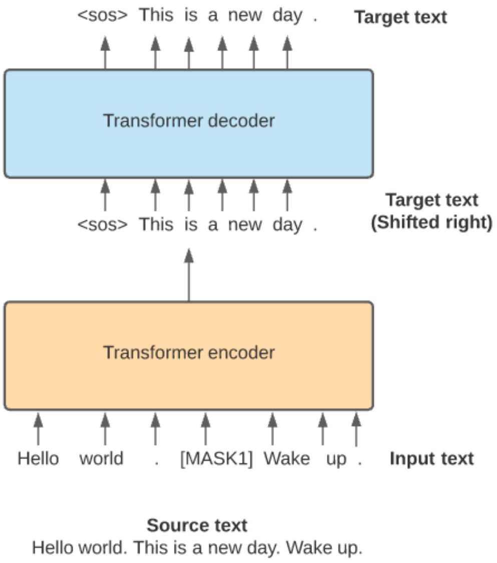
Target

<X> important <Y> NLP

Figure 2.13: T5 input

PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive Summarization) was introduced by Zhang et al. (2020a). PEGASUS is a Transformer model (Vaswani et al., 2017) that is pre-trained with a self-supervised objective.

Inspired by Raffel et al. (2020), Zhang et al. (2020a) proposed a new pre-training objective: GSG (Gap Sentences Generation) (Figure 2.14). It consists of selecting and masking whole sentences from the source documents where each mask label keeps the order of the masked sentence. The masked sentences are concatenated into a pseudo-summary. Furthermore, authors consider three criteria for selecting the sentences to be masked: (1) Random (select m sentences arbitrarily), (2) Lead (select the first m sentences), and (3) Principal (select the m sentences with the highest ROUGE1-F1 score between the sentence and the rest of the document).



<sos> This is a new day .
Target text
Transformer decoder
<sos> This is a new day .
Target text (Shifted right)
Transformer encoder
Hello world . [MASK1] Wake up .
Input text
Source text
Hello world. This is a new day. Wake up.

Figure 2.14: PEGASUS with an example of GSG

PEGASUS was trained with two corpora:

1. *C4* (Raffel et al., 2020) - It consists of texts from 350 millions of Web-pages (750GB)
2. *HugeNews* - It consists of 1.5 billions of news articles (3.8TB).

Zhang et al. (2020a) compare the performance of two tokenizers: Byte-pair-encoding algorithm (BPE) (Wu et al., 2016, Sennrich et al., 2016) and SentencePiece Unigram algorithm (Unigram) (Kudo, 2018). They evaluated Unigrams with different vocabulary sizes ranging from 32k to 256k. Best results were obtained with a vocabulary size of 96K. Besides, Unigrams overcome BPE on datasets are not news.

BART is a sequence-to-sequence model introduced by Lewis et al. (2019), and inspired by Vaswani et al. (2017). It is based on a bidirectional encoder and an auto-regressive decoder. BART also masks some randomly-chosen tokens in the input document (Figure 2.15). We can visualize BART as BERT in the encoder because it is bidirectional, and GTP (Radford et al., 2018) model in the decoder because it is from left to right.

The pre-training of BART incorporates two stages:

1. Some noise is introduced in the text using a noising-function.
2. The model is trained to rebuild the source text.

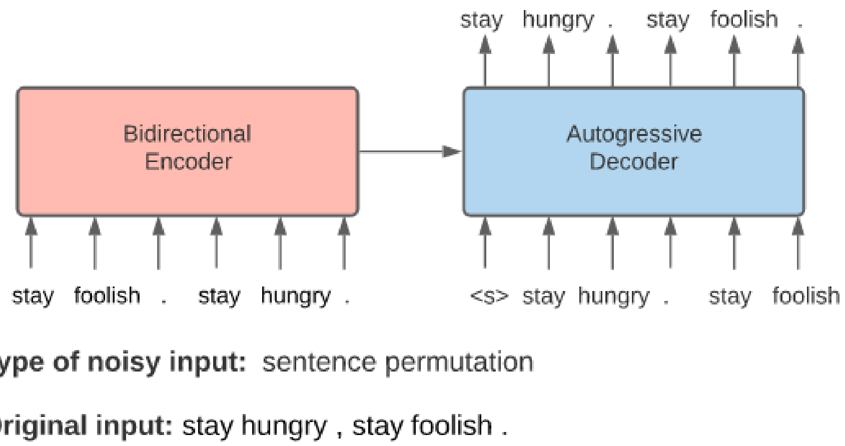
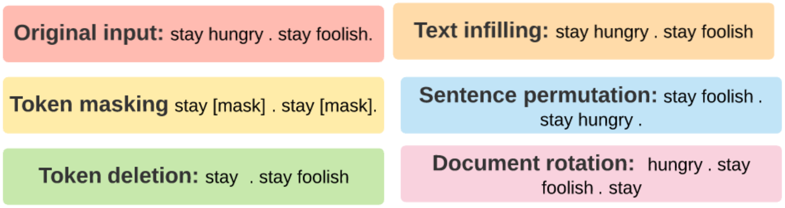

stay hungry . stay foolish .
Bidirectional Encoder
Autogressive Decoder
stay foolish . stay hungry .
<s> stay hungry . stay foolish
Type of noisy input: sentence permutation
Original input: stay hungry , stay foolish .

Figure 2.15: BART architecture

The authors of BART proposed some noising functions that improve the quality of summaries. For instance, token masking, token deletion, text infilling, sentence permutation, document rotation. Figure 2.16 provides examples of such functions.



Original input: stay hungry . stay foolish.

Text infilling: stay hungry . stay foolish

Token masking stay [mask] . stay [mask].

Sentence permutation: stay foolish . stay hungry .

Token deletion: stay . stay foolish

Document rotation: hungry . stay foolish . stay

Figure 2.16: Types of noisy inputs in BART

Once the model is pre-trained, we can fine-tune BART on different downstream applications such as sequence classification tasks, sequence generation tasks, and machine translation.

Hierarchical Transformer was proposed by Liu and Lapata (2019b). It is based on their previous system (Liu et al., 2018) that contains two main components: (1) an extractive summarizer to get the most relevant passages from the source text, and (2) an abstractive summarizer that takes the output of the extractive summarizer and generates the final summary. The improvement brought by the latter consists in learning latent dependencies among text units to help to share hierarchical information between documents. This is achieved using an attention mechanism that helps to concatenate text spans and process them as a flat sequence.

The disadvantage of the hierarchical Transformer is that its performance is heavily dependent first on the quality of the extractive summarizer that guides the search space of the abstractive summarizer and second on the quality of graphs used to capture relationships between text units. Contrarily to hierarchical Transformer, our proposed method in Chapter 4 is end-to-end and entirely Transformer-based.

Reformer was proposed by Kitaev et al. (2020). It is an improvement of the basic Transformer architecture to reduce their memory footprint and computational time. A reformer neural network is mainly based on two modifications that improve a transformer's efficiency significantly:

- Replace the dot product attention with locality-sensitive hashing. This improvement reduces the complexity of processing input sequences from $O(L_{input}^2)$ to $O(L_{input}\ log\ L_{input})$, but needs tuning the number of concurrent hashes in order to find the original transformer performance.

- Use reversible residual layers instead of standard residuals, which allows storing the activation only once at the end of the model instead of after each layer of the model. This technique has a negligible impact on the training process compared to the standard Transformer.

To our knowledge, Reformer has not been yet used for text summarization. Note that Reformer, as well as most of the improved versions of Transformers, invoked in Tay et al. (2020) was released in 2020. They are interesting approaches that could be combined with our proposed method presented in Chapter 4. However, we keep this possibility as a perspective for future research since these methods appeared mostly at the end of this thesis.

2.4.6.4 Choice of the best generated sequence in ATS

During the summary generation process, Neural-based approaches use a linear transformation and a Softmax function to convert the model output to a prediction probability distribution over the vocabulary tokens. The goal then is to use this probability distribution to choose the next word to generate.

Many search algorithms can be applied to predict the next word given the already-generated sequence. The most popular ones are:

- **Greedy search -** is the most intuitive way to produce tokens by choosing each time the token having the highest conditional probability. This algorithm has the advantage of running fast since no complicated heuristic is applied to predict the next token. However, its main drawback is the possibility of losing good-quality sequences that start with low-probability tokens and finish with high-probability ones. At the time step t, the next token in a greedy search algorithm is chosen as in Equation 2.33.

  $$x_t = \underset{x \in \mathcal{X}}{\operatorname{argmax}}\ P(x|x_1, ..., x_{t-1}, c) \tag{2.33}$$

  where x is a token from the vocabulary $\mathcal{X}$ and c is a context variable that encodes the input sequence information.

  The final sequence probability in a greedy search is computed as in Equation 2.34:

  $$p = \prod_{t=1}^{L} P(x_t|x_1, ..., x_{t-1}, c) \tag{2.34}$$

  where L is the maximum generated summary length.

  Unfortunately, there is no guarantee that a greedy search will obtain the optimal sequence.

- **Beam search -** (Graves, 2012, Boulanger-Lewandowski et al., 2013, Sutskever et al., 2014) is the most popular search algorithm to predict the next token in the generated sequence. Unlike greedy search, beam search builds *beam size* $= k$ candidate sequences and chooses the best one of them to be the generated summary.

  The algorithm terminates when one of the sequences encounters the end-of-sequence token. However, if it reaches the maximum output length L and no end-of-sequence token is founded, a length penalty rule is applied to favor long sequences. The chosen sequence is the one maximizing the Equation 2.35:

$$\frac{1}{L^{\alpha}}\, log\ P(x_1, ..., x_L) = \frac{1}{L^{\alpha}} \sum_{t=1}^{L} log\ P(x_t|x_1, ..., x_{t-1}, c) \tag{2.35}$$

where α is a hyper-parameter usually set to 0.75.

Note that when the beam size $k = 1$, beam search is equivalent to greedy search. While increasing the value of k improves search results, it increases execution time as well.

We present below an overview of various automatic text summarization approaches from the state of the art.

2.5 Overview of various Automatic Summarization systems

In this section, we describe the main systems developed in both extractive and abstractive summarization. We report in Tables 2.1 and 2.2 the performance in terms of ROUGE of various ATS systems. ROUGE is an automatic evaluation approach that relies on humans intervention based on lexical overlaps between reference and generated summaries. It has many variants, but we only report the most popular ones: ROUGE-N ($N = \{1, 2\}$ is the n-gram size) and ROUGE-L (Longest Common Subsequence). More details about this metric are provided in Subsection 2.6.2. Evaluation is done on datasets described in Section 2.3. Note that both tables follow a descending chronological order.

2.5.1 Extractive Systems

The first automatic systems were focused on the extractive approach. Luhn (1958) was one of the early pioneers to work on automatic summarization. He assumed that the most important words in a summary are repeated most frequently in the source text. He based his research on $TF - IDF$.

Over time, probabilistic metrics surged, such as Markov Models and Hidden Markov Models (HMM). Researchers applied these methods to automatic summarization (Jing and McKeown, 1999, Conroy and O'leary, 2001, Knight and Marcu, 2002, Suneetha and Sameen, 2012). For instance, Jing and McKeown (1999) proposed an algorithm based on HMM that decomposes human-written summary sentences intending to determine the relations between the phrases of a reference summary and those of the original text. Alternatively, Conroy and O'leary (2001) proposed a method for text summarization that considers three features: (1) the sentence's position in the document (using HMM), (2) the number of terms in the sentence, and (3) the probability of each term. This approach aims to compute the overall sentence probability and decide if it belongs to the summary or not.

Meanwhile, Knight and Marcu (2002) focused on sentence compression using PCFG to assign probabilities to a tree, while Erkan and Radev (2004) computed the centrality of each sentence in the text, where a threshold of $TD - IDF$ determines the salience of each term. Word's centrality is related to the centroid set of documents. Mihalcea and Tarau (2004) introduced TextRank, which is a graph-based ranking model. They proposed two unsupervised methods, the first one is keyword extraction, and the second one is sentence extraction.

Nenkova and Vanderwende (2005) proposed SumBasic. It is a summarization system based on word frequency. SumBasic incorporates content selection and reranking depending on the context. Suneetha and Sameen (2012) proposed text summarization based on HMM tagger to identify the key phrases within the Computer Science documents. To evaluate their system, they used cosine, Jaccard, Jaro-Winkler, and Sorenson similarities.

Probabilistic methods evolved into Machine Learning techniques. For instance, Schilder and Kondadadi (2008) worked on query-based multi-document summarization. They used a Support Vector Machine to rank all sentences in the topic cluster for summarization. Aliguliyev (2009) proposed a method based on sentence clustering. According to the content of the cluster, it identifies the most salient sentences. Alternatively, ShivaKumar and Soumya (2015) used clustering to extractively get summaries where cosine similarity is used to generate the documents clusters. Authors start with finding unique tokens. Afterward, they compute each group's score and sort sentence clusters in reverse order of group score. Finally, they pick the best score sentences from each cluster and add them to the summary.

Besides SVMs, researchers also used Naive Bayes methods to get summaries. One of them is Ramanujam and Kaliappan (2016), who extended the Naive Bayes algorithm's application combined with the timestamp approach multi-document automatic summarization. Another approach is based on the sentences' identified features to determine a sentence's relevance. Kumar et al. (2016) used the following features to determine the most important sentences: title/headline words, sentence position, sentence length, term weight, and proper noun.

Hochreiter and Schmidhuber (1997) introduced LSTM (Long-Short Term Memory). With the emergence of these neural networks, automatic summarization systems outperformed scores obtained by probabilistic methods so far. For about twenty years, most developed systems used LSTM and GRU (Gated Recurrent Unit). GRU is a neural network introduced by Cho et al. (2014) and is a modification of the LSTM neural network.

Cheng and Lapata (2016) used LSTM for the extractive summarization of single documents. They used a decoder that chooses output symbols from the document of interest rather than the entire vocabulary. Alternatively, Nallapati et al. (2017) introduced SummaRuNNer (simple recurrent network-based sequence classifier). SummaRuNNer is a system based on a two-layer bidirectional GRU as the basic building block of the sequence classifier. Furthermore, Sinha et al. (2018) used an approach based entirely on data-driven and a feed-forward neural network to get summaries from single documents.

Transformer Neural Networks (Vaswani et al., 2017) have been increasing the scientific community's interest in text summarization. These models have achieved to improve the quality of automatic summaries when using pre-trained models. For instance, some works were inspired by BERT (Devlin et al., 2019) model, such as Zhang et al. (2019b) and Lu and Jin (2020).

Zhang et al. (2019b) introduced HIBERT (HIerarchical Bidirectional Encoder Representations from Transformers). HIBERT aims to learn the representation of a document on unlabeled data. Lu and Jin (2020) proposed ClinicalBertSum whose goal is to fine-tune BERT for medical abstract summarization on PubMed 200k RTC (Dernoncourt and Lee, 2017) dataset. Hence, they fine-tuned on medical notes (ClinicalBERT), on scientific data (SciBERT (Beltagy et al., 2019)), and on BERT-based text summarization model (BertSum (Liu and Lapata, 2019a)).

Corpus	**System**	**R-1**	**R-2**	**R-L**
CNN/DailyMail News (Hermann et al., 2015)	Lu and Jin (2020)	42.98	20.03	39.38
	Zhang et al. (2019b)	42.37	19.95	38.83
	Nallapati et al. (2017)	39.6	16.2	35.3
	Cheng and Lapata (2016)	21.2	8.3	12.0
PubMed 200k (Dernoncourt and Lee, 2017)	Lu and Jin (2020)	33.58	11.87	27.41

Corpus	System	R-1	R-2	R-L
New York Times (NYT) (Sandhaus, 2008)	Zhang et al. (2019b)	49.47	30.11	41.63
DUC 2007 (Over et al., 2007)	Schilder and Kondadadi (2008)	-	11.0	-
DUC 2006 (NIST, 2014)	Schilder and Kondadadi (2008)	-	9.25	-
DUC02 (NIST, 2014)	Sinha et al. (2018)	55.1	22.6	-
	Cheng and Lapata (2016)	47.4	23.0	43.5
	Aliguliyev (2009)	45.65	11.36	-
	Nenkova and Vanderwende (2005)	47.08	-	-
DUC01 (NIST, 2014)	Aliguliyev (2009)	46.65	17.73	-

Table 2.1: ROUGE scores of some extractive systems from the state of the art

2.5.2 Abstractive Systems

The abstractive approach aims to generate automatic summaries that contain words that are not present in the source text. Inspired by Bahdanau et al. (2014), Rush et al. (2015) introduced an approach of abstractive sentence summarization. Their method combined the Attentional RNN Encoder-Decoder model with an entirely data-driven approach.

A year later, Nallapati et al. (2016) proposed a model that consists of a bidirectional GRU-RNN encoder, a uni-directional GRU-RNN decoder, and an attention mechanism over the source-hidden states. Besides, they adapted to this model a Large Vocabulary Trick (LVT). In the same year, Chopra et al. (2016) produced the abstractive summary of an entry sentence. The model uses a convolutional attention-base conditional recurrent neural network. See et al. (2017) introduced Get To The Point (GTTP) system. GTTP is a news hybrid extractive-abstractive summarization system inspired by Nallapati et al. (2016) and Vinyals et al. (2015). It consists of two main parts: a pointer-generator network and a coverage model. Based on this idea, Cohan et al. (2018) used the pointer-generator network idea applied to their model on medical articles. They proposed a model based on LSTM. The model consists of a hierarchical encoder and an attentive discourse-aware decoder.

Paulus et al. (2017) proposed reinforcement-learning-based algorithms on encoder-decoder architecture. The model consists of a bidirectional LSTM encoder and a single LSTM decoder. The authors used intra-attention in the encoder to focus on specific parts of the input sequence. In contrast, the decoder used this attention to check which words have already been generated and avoid repetitions. While most of the researchers used LSTMs and GRUs, Narayan et al. (2018a) introduced an encoder-decoder abstractive model based on convolutional neural network (CNN) blocks. The input length is fixed in such architectures, and the convolutional neural network blocks compute intermediate states. Further, the interaction between tokens and hierarchical layers captures long-range dependences.

Inspired by Transformers neural networks (Vaswani et al., 2017), pre-trained models such as BERT (Devlin et al., 2019) surged. Therefore, researchers developed several automatic summarization systems. For instance, Liu et al. (2018) aimed to generate English Wikipedia articles based on a Transformer architecture they modified. They used only a decoder with local attention and memory-compressed attention to be able to read long sequences. Interestingly, extractive summaries generated by five well-known systems (identify, TF-IDF, TextRank, SumBasic, Cheating) are used as an input for the neural network to generate abstractive summaries.

Liu and Lapata (2019a) worked on BERTSUM, a variant of BERT where authors modified the input sequence to allow as input many sentences. The architecture is an encoder-decoder called BERTSUMABS, where the encoder is the pre-trained BERTSUM, and the decoder is a standard transformer. Alternatively, Hoang et al. (2019) proposed two training processes for the initialization of the pre-trained GTP model (Radford et al., 2018): domain-adaptive training and end-task training.

Meanwhile, Fabbri et al. (2019) proposed a hierarchical model for neural abstractive multi-document summarization using Transformers Neural Networks. The model consists of a pointer-generator network (See et al., 2017), and the Maximal Marginal Relevance (MMR) (Carbonell and Stewart, 1998). Lewis et al. (2019) introduced BART, which is a sequence-to-sequence model based on noisy objectives of the input document in the pre-training stage. These techniques are token masking, token deletion, text infilling, sentence permutation, and document rotation.

Kim et al. (2019) proposed a memory network model called multi-level memory networks (MMN). This model stores information from the source text in different levels, for instance, word-level, sentence-level, paragraph-level, and document-level. MMN uses a multi-layer CNN as the write network. Zhang and Tetreault (2019) developed a method to generate email subjects from the email body. The technique works in two stages: the extractor selects the most relevant sentences, and later the abstractor paraphrases the selected sentences into a subject line.

Most recently, Zhang et al. (2020a) introduced PEGASUS (Pre-training with Extracted Gap-sentences for Abstractive SUmmarization sequence-to-sequence models). PEGASUS is a new pre-training objective using Transformers: GSG (Gap Sentences Generation), which consists of masking whole sentences from a document and generating these gap sentences from the rest of the document.

Corpus	System	R-1	R-2	R-L
arXiv (arXiv, 2021)	Zhang et al. (2020a)	44.70	17.27	25.80
	Cohan et al. (2018)	35.80	11.05	31.80
Multi-News (Fabbri et al., 2019)	Zhang et al. (2020a)	47.52	18.72	24.91
	Fabbri et al. (2019)	43.47	14.89	17.41
Reddit TIFU (Kim et al., 2019)	Zhang et al. (2020a)	26.54	8.94	21.64
	Kim et al. (2019)	20.2	7.4	19.8
BIGPATENT (Sharma et al., 2019)	Zhang et al. (2020a)	53.63	33.16	42.25
AESLC (Zhang and Tetreault, 2019)	Zhang et al. (2020a)	37.69	21.85	36.84
	Zhang and Tetreault (2019)	23.67	10.29	23.44
BillSum (Kornilova and Eidelman, 2019)	Zhang et al. (2020a)	57.31	40.19	45.82
NEWSROOM (Grusky et al., 2018)	Kim et al. (2019)	17.5	4.7	14.2
	Zhang et al. (2020a)	45.15	33.51	41.33
WikiHow (Koupaee and Wang, 2018)	Zhang et al. (2020a)	43.06	19.71	34.80
WikiSum (Liu et al., 2018)	Liu et al. (2018)	-	-	38.8
PubMed (Cohan et al., 2018)	Zhang et al. (2020a)	45.49	19.90	27.69
	Cohan et al. (2018)	38.93	15.37	35.21
XSum (Narayan et al., 2018a)	Lewis et al. (2019)	45.14	22.27	37.25
	Zhang et al. (2020a)	47.21	24.56	39.25
	Liu and Lapata (2019a)	38.81	16.50	31.27
	Kim et al. (2019)	32.0	12.1	26.0
	Narayan et al. (2018a)	31.89	11.54	25.75
CNN/DailyMail News (Hermann et al., 2015)	Lewis et al. (2019)	44.16	21.28	40.90
	Zhang et al. (2020a)	44.17	21.47	41.11
	Liu and Lapata (2019a)	42.13	19.60	39.18
	Paulus et al. (2017)	41.16	15.75	39.08
	See et al. (2017)	39.53	17.28	36.38
	Nallapati et al. (2016)	35.46	13.30	32.65

Corpus	System	R-1	R-2	R-L
Gigaword (Napoles et al., 2012)	Zhang et al. (2020a)	39.12	19.86	36.24
	Chopra et al. (2016)	33.78	15.97	31.15
	Rush et al. (2015)	31.00	12.65	28.34
New York Times (NYT) (Sandhaus, 2008)	Liu and Lapata (2019a)	49.02	31.02	45.55
	Paulus et al. (2017)	47.22	30.51	43.27
DUC 2004 (NIST, 2014)	Chopra et al. (2016)	28.97	8.26	24.06
	Rush et al. (2015)	28.18	8.49	23.81
DUC 2003 (NIST, 2014)	Fabbri et al. (2019)	35.78	8.90	11.43
	Nallapati et al. (2016)	28.61	9.42	25.24

Table 2.2: ROUGE scores of some abstractive systems from the state of the art

In the following section, we present the most popular automatic summary evaluation approaches from the state of the art.

2.6 Methods for Summary Evaluation

Evaluation methods are fundamental techniques to assess if summaries generated by an automatic system capture the original document's idea. Different evaluation methods have been developed in the last decade for the evaluation of automatically-generated summaries. It exists two types of evaluation methods: (1) manual evaluation methods like Pyramid (Nenkova and Passonneau, 2004) and Responsiveness, where participation of human is mandatory, and (2) automatic evaluation methods (Lin, 2004, Torres-Moreno et al., 2010, Cohan and Goharian, 2016, Cabrera-Diego and Torres-Moreno, 2018), where the presence of reference summaries generated by humans is not compulsory.

2.6.1 Manual evaluation methods

2.6.1.1 Precision and Recall

These two well-known metrics can be used to evaluate extractive summaries. *Precision* and *Recall* compare summaries generated by automatic systems with those generated by humans (goal standards) and compute lexical overlap.

Precision is the fraction of correct system sentences (Nenkova, 2006):

$$Precision = \frac{|system - human\ choice\ overlap|}{|sentences\ chosen\ by\ system|} \tag{2.36}$$

Recall is the fraction of sentences chosen by the human that were also correctly identified by the system (Nenkova, 2006):

$$Recall = \frac{|system - human\ choice\ overlap|}{|sentences\ chosen\ by\ human|} \tag{2.37}$$

According to Nenkova (2006), Precision and Recall comprise many drawbacks such as:

- **Human variation -** Since humans select the sentences, they can be highly subjective, and many humans can select different sentences.
- **Granularity -** The sentences can be of different lengths, leading to information granularity variation.
- **Semantic equivalence -** Two sentences can be written with different words and have the same meaning.

2.6.1.2 Relative Utility

Relative Utility (RU) (Radev et al., 2003) is a method for evaluating single and multi-document extractive summaries. It compares sentence selection between the summaries generated by automatic systems and reference summaries. This method can optionally penalize summaries that contain sentences with redundant information. RU assigns numerical scores to individual sentences.

RU has shown better evaluation results compared to other methods like Precision, Recall, Percent Agreement (PA) (Owczarzak et al., 2012) and Kappa (Carletta, 1996). However, it is not suitable to distinguish between human-written and automatic summaries.

2.6.1.3 DUC Manual Evaluation

DUC is an NLP annual challenge where researchers attempt to solve one or multiple tasks, including automatic summarization.

The goal of DUC is to compile standard training and test collections that can be shared among researchers and provide standard and large-scale evaluations in single and multiple document summarization for their participants (Lin and Hovy, 2002).

The first DUC challenge was lead in 2001, and it included three tasks: (1) fully automatic single-document summarization, (2) fully automatic multi-document summarization, and (3) exploratory summarization. The main idea behind DUC's summarization task is to produce automatic summaries and compare them to those written by humans.

Human summarization was initially done by selecting the most important sentences in the text. Nowadays, DUC relies on human abstracts as gold standard models. Abstractive summarization is more complex than extractive summarization because the text is paraphrased (Nenkova, 2006).

2.6.1.4 Pyramid

Manual evaluation with Pyramid (Nenkova and Passonneau, 2004) is based on Summary Content Units (SCUs). SCUs are groups of sub-parts of sentences taken from several reference summaries, representing at most one clause and sharing the same meaning. SCUs are weighted depending on the number of reference summaries they are found in, and a candidate summary score is computed using the weights of its SCUs.

The process of evaluation with Pyramid begins with the identification of similar sentences in other summaries. After that, the sub-parts of these sentences are manually studied in detail to get the SCUs. Note that each content unit has a unique index, weight, and label.

We present below an example of four sentences:

1. World War II, or **Second World War was a global war from 1939 to 1945**.
2. **The Second World War started in 1939**. It marked between 70 and 85 million deaths.
3. **Second World War began on September 1, 1939**, with the invasion of Poland by Germany.
4. **Second World War finished on September 2, 1945**.

In this example, we identify three content units: SCU1 = Second World War (weight=*4*), SCU2 = 1939 (weight = *3*), and SCU3 = 1945 (weight= *2*). The total number of levels in the Pyramid is four because the maximum weight identified is four. At a given level, the Pyramid can contain more than one SCU. For instance, in the lowest level (content units frequency is equal to 1) there are words such as *started*, *began* and *finished*. The Pyramid score ranges from 0 to 1. The score is computed by dividing the sum of the SCUs weights by an optimal weights sum with the same number of SCUs.

The advantage of Pyramid is that scores are stable over annotators and are high for human summaries. However, the Pyramid metric is very costly due to the necessity of manual annotations at the sub-clause level.

2.6.1.5 PyrEval

Over the years, other variants of Pyramid have surged. For instance, Gao et al. (2019) proposed an automated Pyramid called PyrEval. It produces human-readable pyramids. The first PyrEval step consists of decomposing sentences into segments from the reference summaries using Stanford Core NLP (Manning et al., 2014). In the second step, these segments are converted to semantic vectors. The third step uses the EDUA (Emergent Discovery of Units of Attraction) algorithm to get an optimal pyramid by maximizing the semantic similarity of the segments to get the SCUs. Finally, PyrEval makes use of WMIN (Shuichi et al., 2003) to find the matches between candidate summaries and the SCUs.

2.6.1.6 LitePyramid

Proposed by Shapira et al. (2019), is a crowdsource-based lightweight version of Pyramid (Nenkova and Passonneau, 2004). It emulates the two Pyramid phases: pyramid creation and system evaluation. In the first phase, LitePyramid relies on many reference summaries. However, unlike the pyramid, it guides two crowd workers to extract 8 SCUs per reference summary, leading to 16 SCUs per reference summary. After filtering long sentences, LitePyramid keeps 13 SCUs per reference summary. In the second phase, a crowd-worker is presented with a system summary and a fixed set of SCUs, where the candidate summary score is the percentage of SCUs it matched among the set of judged SCUs.

2.6.2 Automatic evaluation methods with human references

2.6.2.1 ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) was proposed by Lin (2004). ROUGE is the most influential method to evaluate automatic summaries. It is based on word overlap between reference summaries and a candidate summary. In the following, we describe different ROUGE variants.

ROUGE-N (Lin, 2004) is related to the recall between the candidate summary and reference summaries. In general, the N values are *1,2* and *3*. We call these values unigram, bigram, and trigram, respectively.

$$\text{ROUGE-N} = \frac{\sum_{S \in RS} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in RS} \sum_{gram_n \in S} Count(gram_n)} \tag{2.38}$$

where:

- RS are the reference summaries
- n is the n-gram size
- $Count_{match}(gram_n)$ is the maximum number of n-grams co-ocurring in a candidate summary and set of reference summaries.

The equation above computes ROUGE-N using one reference summary. The following equation computes it using multiple references:

$$\text{ROUGE-N}_{multi} = argmax_i(\text{ROUGE-N}\{r_i, s)\} \tag{2.39}$$

where:

- s is a candidate summary
- r_i is every reference summary in RS

ROUGE-L (Longest Common Subsequence). Given two word sequences (X and Y), ROUGE-L searches for the longest common sub-sequence of X in Y. We assume that Y is larger than X.

ROUGE-W (Weighted Longest Common Sub-sequence) is an improved version of ROUGE-L, where the sequence words can be consecutive or not (separated by intermediate words). ROUGE-W keeps control over the size of the consecutive terms.

ROUGE-S (Skip-Grams) measures the overlapping of skip-grams between the candidate summary and reference summaries. A skip-gram is an ordered pair of words in a sentence that allows an arbitrary gap.

ROUGE-SU (Skip-Unigrams) is an improved version of ROUGE-S that does not consider the candidate sentences if they do not contain a skip-gram. ROUGE-SU takes into account the unigrams in the evaluation.

2.6.2.2 WE-ROUGE

It is an improved version of ROUGE proposed by Ng and Abrecht (2015) for abstractive summary evaluation. Authors integrate word embeddings obtained with Word2Vec into ROUGE in order to handle its bias towards lexical similarities. WE-ROUGE (Word Embeddings ROUGE) uses word embeddings instead of raw text to compute the semantic similarity of words between candidate and reference summary where a zero value is attributed if one of the compared words is OOV (Out-Of-Vocabulary). To handle n-gram OOVs, authors compose individual word embeddings with the multiplicative approach from Mitchell and Lapata (2008).

This method is interesting insofar as it replaces syntactic tokens with word embeddings for semantic representations. However, it is still dependent on the quality of the model used to get the word embeddings.

2.6.2.3 SERA

SERA (Summarization Evaluation by Relevance Analysis) (Cohan and Goharian, 2016) is based on a content relevance analysis between a candidate summary generated by an ATS system and reference summaries written by humans (at least one). For this, a search engine for information retrieval is used. The search engine takes as input: (1) a set of documents to index related to the candidate summary topic, and (2) queries, which are a candidate summary and its corresponding reference summaries. In SERA, queries can be reformulated in three ways:

- **Raw text** - only stop words and numbers are removed
- **Noun phrases** (NP) - only noun phrases are kept while other words are deleted
- **Keywords** (KW) - only unigrams, bigrams, and trigrams are kept

An overview of the method is presented in Figure 2.17. SERA searches for the queries in the index and provides a list of documents ranked according to their similarity with the queries. A candidate summary score is then the similarity between the lists of retrieved documents, both truncated at a given point. Thus, two summaries are similar if they are related to the same set of documents (Equation 2.40).

$$SERA = \frac{1}{M} \sum_{i=1}^{M} \frac{|R_C \cap R_{G_i}|}{|R_C|} \tag{2.40}$$

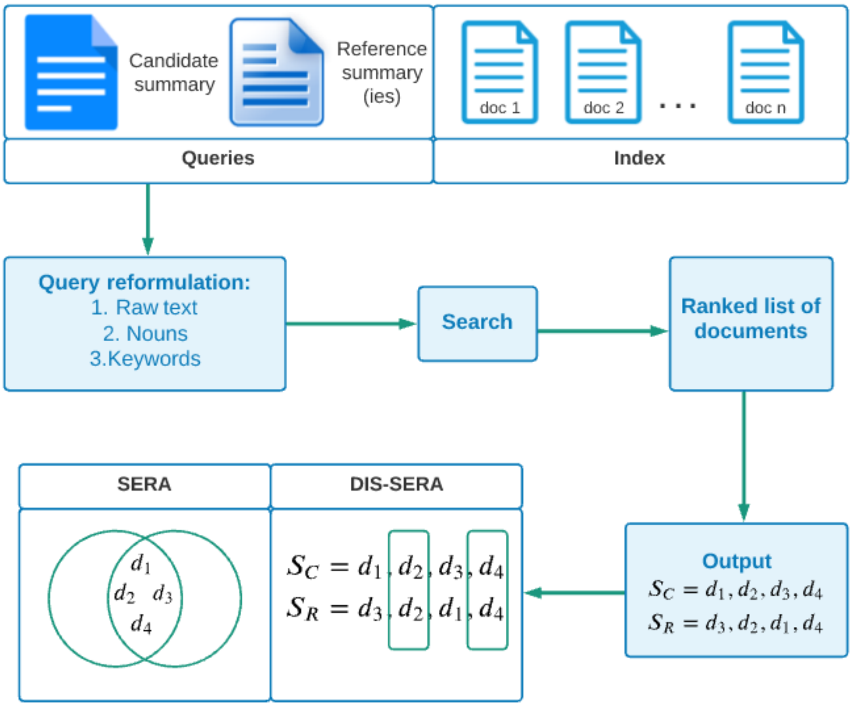


Figure 2.17: An overview of SERA evaluation approach

Where: R_C is the ranked list of retrieved documents for the candidate summary C, R_{G_i} is the ranked list of retrieved documents for the gold summary G_i, and M is the number of reference summaries. We assume that $|R_C| \geq |R_{G_i}|$.

As shown in Equation 2.40, SERA is based on an intersection between the two sets of documents related to the queries. The authors of SERA propose SERA-DIS, a variant of SERA that considers the order of related documents (Equation 2.41).

$$SERA - DIS = \frac{\sum_{i=1}^{M}(\sum_{j=1}^{|R_C|}\sum_{k=1}^{|R_{G_i}|} X_{j,k}}{M * D_{max}}$$

$$X_{j,k} = \begin{cases} \frac{1}{log(|j-k|+2)} & if\ R_C^{(j)} = R_{G_i}^{(k)} \\ 0 & otherwise \end{cases} \tag{2.41}$$

Where: $R_C^{(j)}$ is the j^{th} result in the ranked list R_C, and D_{max} is the maximum achievable score used as a normalization factor.

In both SERA variants, retrieved results are truncated at 5 and 10 documents (Hence the notations SERA-5 and SERA-10 in Section 3.3).

2.6.2.4 BERTScore

Proposed by Zhang et al. (2020b), BERTScore is a language generation evaluation approach based on contextual embeddings extracted with the BERT model. It was mainly designed for machine translation at the sentence level and image captioning but can be adapted to summary evaluation. Evaluation with BERTScore is done using a greedy matching of cosine similarity between candidate and reference summaries' embeddings. The matching consists of relating each token in the candidate summary with the most similar token in the reference summary. The advantage of using contextual embeddings is that a different embedding is attributed to a word depending on its context. This way of processing is more flexible and robust compared to exact-string (Papineni et al., 2002) or heuristic (Lavie and Agarwal, 2007) matching.

Unlike ROUGE (Lin, 2004) that is based on lexical overlaps, BERTScore makes use of contextual embeddings that are effective for paraphrase detection. An advanced version of BERTScore adds a weighting term obtained by computing the *Inverse Document Frequency* (IDF) to give more importance to rare words that are more indicative for sentences similarity (Vedantam et al., 2015).

2.6.2.5 SSAS

SSAS (*Semantic Similarity for Abstractive Summarization*) was proposed by Vadapalli et al. (2017). As BERTScore (Zhang et al., 2020b), it is based on semantic matching between candidate and reference summaries. First, SCUs are extracted automatically from candidate and reference summaries using the PEAK model from Yang et al. (2016). Second, a set of NLP inferences and paraphrasing features are applied. Authors compute a weighted composition to leverage scores from different measures in a single normalized score.

Used measures include (1) Combined Entailment Scores, (2) Combined Contradiction Scores, (3) Combined Topic Neutrality Scores, (4) Paraphrasing probabilities using the model from Kiros et al. (2015) and (5) ROUGE-SU4 scores. SSAS showed competitive performance compared to previous abstractive summary evaluation methods. However, it is computationally very expensive because of the large number of semantic models used to compute features.

2.6.2.6 MoverScore

Proposed by Zhao et al. (2019), this evaluation method combines contextualized representations with *Earth Mover Distance* (EMD) from Rubner et al. (2000). The latter measures the travelling distance of moving from the word frequency distribution of the candidate summary to that of the human-written one. Authors explore two variants of this distance: (1) word mover (Kusner et al., 2015) and (2) sentence mover. The difference between both is in the granularity of comparison between embeddings. According to the authors, contextual representations are beneficial to encode both syntactic and semantic deviations between candidate and reference summaries. Once again, MoverScore is also time-expensive insofar as it is based on fine-tuning the BERT model on three Natural Language Inference (NLI) datasets before being used for evaluation.

2.6.3 Automatic evaluation methods without human references

Although gold standards are solid reference frames, summary evaluation using gold standards is costly and time expensive. The subjectivity of humans when summarizing articles makes evaluation heavily dependent on their expertise domain. For these reasons, researchers are looking for efficient alternatives to evaluate automatic summaries without any human intervention. One of the proposed solutions to assess the content selection of systems is based on three features: distributional similarity, summary likelihood, and topic words in the summary. Judgments of linguistic quality join these features. These approaches led to lower correlations with responsiveness than the content-based pyramid evaluation (Louis and Nenkova, 2009). Very recently, other sophisticated automatic methods were born and are described below.

2.6.3.1 SummTriver

SummTriver (ST) (Cabrera-Diego and Torres-Moreno, 2018) is an evaluation metric that does not need any human intervention (model summaries). Instead, it computes the trivergence between three probability distributions (R, P, and Q), where:

- R is the probability distribution generated by the summary to evaluate.
- P is the probability distribution of a set of summaries that are different from R but share the same source document.

- Q is the probability distribution of the source document from which R and P were obtained.

The trivergence is a combination of different divergences and is computed in two ways:

- as a composition of two divergences:

$$\mathcal{T}_c(P \mid\mid Q \mid\mid R)) = d\left(P \mid\mid \frac{d(Q \mid\mid R)}{N}\right) \tag{2.42}$$

  where d is a divergence and N is a normalization parameter.

- as a multiplication of three divergences :

$$\mathcal{T}_m(P \mid\mid Q \mid\mid R)) = d(P \mid\mid Q) \cdot d(P \mid\mid R) \cdot d(Q \mid\mid R) \tag{2.43}$$

The authors used two types of divergences: Kullback-Leibler (KL) and Jensen-Shannon (JS), such that:

Kullback-Leibler divergence measures the dissimilarity of two probability distributions over the same event space and is defined as:

$$KL(P \mid\mid Q) = \sum_{\omega \in P} p_P(\omega)\ log_2 \frac{p_P(\omega)}{p_Q(\omega)} \tag{2.44}$$

where:

- ω is an event
- $p_P(\omega)$ is the probability of event ω in distribution P
- $p_Q(\omega)$ is the probability of the same event but in distribution Q

Note that KL divergence is asymmetric, and authors use its smoothed version to handle unseen events.

Jensen-Shannon divergence measures the dissimilarity of two probability distributions using their mean and is defined as:

$$JS(P \mid\mid Q) = \frac{1}{2}\ KL(P,\ M) + \frac{1}{2}\ KL(Q,\ M) \tag{2.45}$$

where $M = \frac{1}{2}\ (P + Q)$.

Note that this divergence is symmetric, and thus both its smoothed and non-smoothed versions are applicable.

2.6.3.2 FRESA

FRESA (Torres-Moreno et al., 2010) (*FRamework for Evaluating Summaries Automatically*)[1] is a multilingual evaluation system that directly compares the candidate summary with its source document. FRESA works in French, Spanish, English, and German. SummTriver is also based on KL, JS, and sJS divergences to determine the summary quality. Furthermore, different kinds of n-grams can be used to compute divergences. Equation 2.46 describes the Jensen-Shannon divergence (Lin et al., 2006) used in FRESA system.

$$Q_w == \begin{cases} & P_\omega = \frac{C^T_\omega}{N} \\ \frac{C^S_\omega}{N_S} & \text{if } \omega \in \text{S} \\ \frac{C^T_\omega+\delta}{N+\delta*B} & \text{otherwise} \end{cases} \tag{2.46}$$

where:

- P is the probability distribution of the words w in text T
- Q is the probability distribution of words w in summary S
- N is the number of words in the text and in the summary $N = N_T + N_S$
- $B = 1.5|V|$
- C^T_ω is the number of words in the text
- C^S_ω is the number of words in the summary

2.6.3.3 SUM-QE

This approach was proposed by Xenouleas et al. (2019). It adapts Quality Estimation (QE) from machine translation to summary evaluation without human references. This approach focuses on linguistic quality such as non-redundancy, referential quality, structure, and coherence. As BERTScore (Zhang et al., 2020b), a BERT model is used to get word embedding, while a linear regression model predicts the summary's quality score. Three versions of SUM-QE were proposed depending on how BERT is fine-tuned.

[1] `http://fresa.talne.eu`

2.6.3.4 End-to-end SQA

Proposed by Bao et al. (2020), End-to-end SQA (*End-to-end Semantics-based Summary Quality Assessment*) is a deep-learning-based approach for summary evaluation without human references. This method is based on two main stages:

- A deep model is trained on a summarization task (CNN/DailyMail, Newsroom, and Big-Patent), where the model's input is a concatenation of the word embedding vectors of the document and candidate summary, while the model's output is a score telling how much the summary is similar to its source document. To train the model in a supervised fashion, the authors generate negative summaries in two ways: (1) by mutating randomly chosen tokens in the gold-standard summary, and (2) by cross pairing summaries between documents. As in SUM-QE (Xenouleas et al., 2019), the BERT model was the one achieving the best results.

- Once the deep model is trained, participants' summaries from TAC 2010 are passed through the network. The output score is used to compute correlation with human evaluations in terms of linguistic quality, modified score, and overall score.

The disadvantage of deep-learning-based approaches is the significant amount of time needed to train the network and perform inference.

2.7 Conclusion

In this chapter, we presented the most important approaches in automatic text summarization and automatic summary evaluation. The latter is essential insofar as we need to assess the quality of generated summaries in order to be able to compare and improve different summarization systems.

The most popular evaluation method used by the scientific community is ROUGE (Lin, 2004), a lexical-based approach. When the research effort moved along abstractive summarization, ROUGE became unfair to evaluate abstractive summaries since they do not necessarily contain tokens from reference summaries. For this reason, the scientific community has proposed methods to evaluate summaries automatically. Some systems do not need reference summaries, such as FRESA (Torres-Moreno et al., 2010), and SummTriver (Cabrera-Diego and Torres-Moreno, 2018). However, they achieve lower correlations with human judgments compared to measures that rely on human references. For instance, SERA (Cohan and Goharian, 2016) is a metric based on content relevance analysis between a candidate summary and a set of reference summaries. SERA achieved better correlations with human evaluations than ROUGE in the medical domain. Inspired by this research, we propose wikiSERA (Chapter 3). This open-source system evaluates summaries belonging to the general domain, and it achieves better correlations with manual approaches than SERA in most of the tested configurations.

Note that some evaluation methods such as End-to-end SQA (Bao et al., 2020), BERTScore (Zhang et al., 2020b), MoverScore (Zhao et al., 2019) and SUM-QE (Xenouleas et al., 2019) appeared very recently. In contrast, works such as SSAS (Vadapalli et al., 2017) are computationally very expensive due to the use of multiple deep learning models at once. Thanks to the system scores provided by Bhandari et al. (2020), we could compare BERTScore and MoverScore with our method proposed in Chapter 3, but we do not involve the other methods for comparison in this thesis.

Concerning automatic summarization, we studied the evolution of ATS methods while focusing on each system's merits and limitations. While the first summarization method developed by Luhn (1958) was based on a frequency-based approach, the following methods used probabilistic methods, machine learning approaches, and deep learning. According to the literature, deep learning techniques nowadays achieve the highest scores in the ATS task, especially Transformer neural networks (Vaswani et al., 2017). We choose to use these neural networks because we are interested in summarizing long medical texts. In Chapter 4, we present $HazPi$, a multi-encoder transformer that aims to produce summaries from long medical texts.

Chapter 3

Automatic Evaluation of general-domain summaries

3.1 Introduction

Text summarization has gained lots of attraction in the last decade. Many approaches have been proposed to generate automatic text summaries, especially neural-based abstractive ones (Chopra et al., 2016, Nallapati et al., 2016, See et al., 2017, Liu and Lapata, 2019a). However, automatic summary evaluation is as crucial as its summarization. To generate summaries of good quality, we need to assess their quality in order to improve summarization systems (Lin, 2004, Torres-Moreno et al., 2010, Cabrera-Diego and Torres-Moreno, 2018). Summaries generated by humans are the best reference to evaluate summaries generated automatically (Lin and Hovy, 2002). However, manual summarization is costly and time-consuming. Equally, summary quality can be biased by the expert opinion, leading to subjective summaries (Lin and Hovy, 2002).

Researchers have developed various methods to evaluate automatic summaries (Cabrera-Diego and Torres-Moreno, 2018, Lin and Hovy, 2002, Lin, 2004, Torres-Moreno et al., 2010). According to Sparck Jones and Galliers (1996), summaries can be evaluated either extrinsically or intrinsically. On the one hand, extrinsic evaluation methods assess summaries depending on their effect on a specific task. They can be done by humans or automatic systems. On the other hand, intrinsic evaluation approaches assess summaries against gold standard summaries and can be manual or automatic. Furthermore, automatic text summarization can be mono- or multi-document based (Aries et al., 2019). Mono-document based summarization systems take as input one document and produce one summary. However, multi-document based summarization systems take as input a set of documents (called a topic) and produce one summary shared by these documents. In this thesis, we are interested in intrinsic evaluation of mono-document, and multi-document based summaries; we will explain the most relevant manual and automatic metrics of summary evaluation. In manual evaluation, human intervention is mandatory, while in automatic evaluation, humans possibly participate in the evaluation process; some evaluation approaches need human intervention while others do not.

The most popular manual methods are Pyramid (Nenkova and Passonneau, 2004) and Responsiveness. The Pyramid approach starts with detecting SCUs (Summary Content Units, defined in Subsection 2.6.1.4 of Chapter 2) in the human reference summaries. The weight of each SCU is its frequency. Later, the human judges search for the same SCUs in the summaries to evaluate, and a score is attributed to each summary depending on the weight of its SCUs. Responsiveness evaluates the content and linguistic quality of automatic summaries. The most popular automatic evaluation metric is ROUGE (Lin, 2004). ROUGE relies on human references and provides a high correlation with manual methods, mostly in extractive summarization. ROUGE is based on the lexical overlap between tokens and phrases in reference summaries and the generated one (Cohan and Goharian, 2016, Lu and Jin, 2020). We also find AutoSummENG and MeMoG (Giannakopoulos and Karkaletsis, 2011), two automatic evaluation approaches with human references based on n-gram graphs. AutoSummENG and MeMoG are statistically equivalent to ROUGE (Cabrera-Diego et al., 2016), and they are highly correlated with manual measures like Pyramid. However, build graphs is expensive in terms of computation. The automatic evaluation of summaries without human references assesses the summary generated by an automatic system against its source document(s), instead of a gold summary. Among these systems, we find FRESA (Torres-Moreno et al., 2010) and SummTriver (Cabrera-Diego and Torres-Moreno, 2018) defined later.

The vast advancements in NLP helped in migrating automatic summarization from extractive to abstractive. In such protocols, ROUGE (Lin, 2004) fails to assess the quality of abstractive summaries since it is possible to have a summary of good quality that expresses the main idea of the document using only the essential terms that occurred in it. Abstractive summarization requires new evaluation metrics. They do not heavily depend on the lexical content of documents. SERA (*Summarization Evaluation by Relevance Analysis*) (Cohan and Goharian, 2016) was proposed as an alternative to ROUGE in the biomedical domain. It focuses on the documents' semantic content, leading to efficiently assess the quality of summaries that are lexically different but express the same idea.

ROUGE (Lin, 2004) was used as a benchmark in DUC 2001-2003 and TAC 2001 challenges to prove its effectiveness in evaluating automatic summarization, while the performance of SERA (Cohan and Goharian, 2016) was tested with TAC 2014 dataset that contains medical articles. SERA shows a better correlation with human summaries than ROUGE since it is based on text content relevance. SERA would effectively evaluate both abstractive and extractive summaries, while ROUGE could fairly evaluate extractive summaries only. Researchers have tested SERA in the biomedical domain. In this thesis, we take out SERA from the biomedical to the general domain and test it with TAC 2008, TAC 2009, and CNNDM (Bhandari et al., 2020) datasets. Besides, we propose wikiSERA, a new SERA version with refined queries, based on the analysis of various corpora containing biomedical and news texts.

3.2 Proposed approach: wikiSERA

SERA (Cohan and Goharian, 2016) was initially proposed for biomedical summary evaluation, where it achieved a better correlation with Pyramid (Nenkova and Passonneau, 2004) than ROUGE (Lin, 2004) on the PubMed dataset (Table 3.1). We hypothesize that it is possible to take out SERA from the biomedical domain to the general one. Therefore, we led an experiment that consists of computing POS Tags distribution percentages for (1) PubMed, a biomedical dataset built by Cohan et al. (2018), (2) AQUAINT-2 (*Advanced Question-Answering for Intelligence*), a dataset specialized in news, and (3) Wikipedia, a general-domain encyclopedic dataset. Figure 3.1 shows bar plots for Nouns, Verbs, Adjectives (Adj.), Prepositions (Prep.), and others.

Metric	Pearson	Spearman	Kendall
ROUGE-3-F	**0.878**	0.841	0.69
SERA-NP-5	0.859	**1.0**	**1.0**

Table 3.1: Best SERA and ROUGE results from Cohan and Goharian (2016) in terms of Pearson, Spearman, and Kendall on the PubMed biomedical corpus

According to Kieuvongngam et al. (2020), nouns represent more accurate information in the generated summaries than the original abstracts, which in our hypothesis explains why Cohan and Goharian (2016) achieves a higher correlation than ROUGE against human evaluations. However, our analysis of three datasets belonging to different domains shows that the distribution of verbs and adjectives is higher in AQUAINT-2 (news) and Wikipedia (general domain) than PubMed

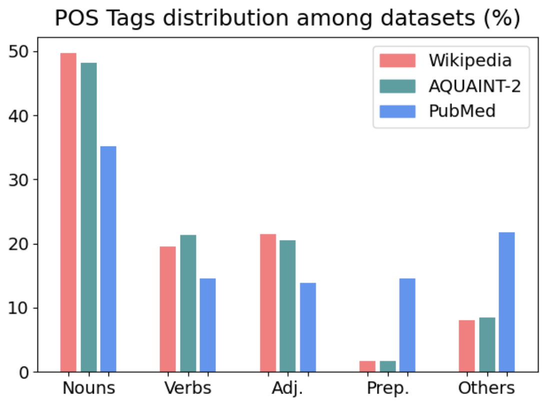


Figure 3.1: POS Tags distribution percentages for Wikipedia, AQUAINT-2, and PubMed datasets

(biomedical domain) dataset. Besides, we noticed a remarkable absence of prepositions from Wikipedia and AQUAINT-2 datasets. Based on this analysis, we proposed wikiSERA, which redefines queries by considering the three most frequent tags in the news and the general domain corpus: nouns, verbs, and adjectives.

According to Cohan and Goharian (2016), SERA was developed in the context of scientific biomedical article summarization with the idea that its semantic specificity is particularly useful. We drop this assumption and hypothesize that wikiSERA can assess summaries from other domains for both abstractive and extractive summarization. This hypothesis is based on the fact that SERA is a measure that considers terms that are not lexically equivalent but are semantically related, thanks to the underlying search engine matching algorithms. We conduct extensive experiments on both SERA and wikiSERA to prove our hypothesis. Results show that wikiSERA achieves better correlations with Pyramid (Nenkova and Passonneau, 2004) than SERA and even outperforms ROUGE in some cases.

3.3 Experiments

3.3.1 Baselines

In the following, we describe some of the most influential evaluation metrics that we used as baselines for evaluation: (1) ROUGE and SERA, two automatic evaluation approaches that rely on human intervention, (2) SummTriver and FRESA, two entirely automatic evaluation metrics, and (3) BERTScore and MoverScore, two BERT-based automatic approaches that rely on human intervention (Devlin et al., 2019). We also provide results with a simple Jensen-Shannon baseline used in Bhandari et al. (2020).

- **ROUGE** (Lin, 2004) - (Recall-Oriented Understudy for Gisting Evaluation) is a measure for evaluating automatic summaries that rely on human gold-standard summaries. It is inspired by the successful evaluation method BLEU (Papineni et al., 2002) in machine translation and is based on lexical overlaps. As explained in Chapter 2, there are different variants of ROUGE, but we only report results with the most popular ones: ROUGE-1, ROUGE-2, and ROUGE-L.

- **SERA** (Cohan and Goharian, 2016) - Because an idea can be expressed in different ways, abstractive summaries do not necessarily contain words that are present in the text. In such cases, ROUGE scores drop since the latter is based on lexical overlaps. SERA was proposed to overcome this issue by giving more importance to the semantic content of summaries. SERA is based on content relevance analysis between a candidate summary and its corresponding reference summaries using information retrieval. This evaluation approach is explained in detail in Subsection 2.6.2.3 of Chapter 2.

- **SummTriver** (Cabrera-Diego and Torres-Moreno, 2018) - is an automatic evaluation method that does not need any reference summary. It is based on trivergence between the source document(s), the candidate summary, and a set of other candidate summaries generated by other summarization systems. Trivergence is computed in two ways: as a composition of two divergences ($\mathcal{T}_c$) or as a multiplication of three divergences ($\mathcal{T}_m$). Three kinds of divergences are used: Kullback-Leibler (KL), Jensen-Shannon (JS), and smoothed Jensen-Shannon (sJS) divergences. The combination of parameters results in the following SummTriver variants: ST-JS-$\mathcal{T}_m$, ST-sJS-$\mathcal{T}_m$, ST-KL-$\mathcal{T}_m$, ST-JS-$\mathcal{T}_c$, ST-sJS-$\mathcal{T}_c$, and ST-KL-$\mathcal{T}_c$.

- **FRESA** (Torres-Moreno et al., 2010) - *(FRamework for Evaluating Summaries Automatically)* also does not need human intervention and is based on the divergence between the source document and the candidate summary. FRESA has five variants: uni-grams (FRESA-1), bi-grams (FRESA-2), tri-grams (FRESA-3), and SU4 (FRESA-4). Since FRESA is basically designed for mono-document evaluation, we concatenated all the articles on the same topic to run it on TAC 2008 and TAC 2009.

- **BERTScore** (Zhang et al., 2020b) - is a metric based on contextual embeddings representation. It needs a candidate summary and at least a reference summary. BERTScore uses contextual embeddings to represent the text's tokens. It computes the matching between the candidate and the reference summary through cosine similarity.

- **JS**-2 (Lin et al., 2006) - *Jensen-Shannon divergence* between bi-gram's distribution of the candidate and reference summaries. This metric is described in Subsection 2.6.3.2 of Chapter 2.

- **MoverScore** (Zhao et al., 2019) - combines contextualized embeddings extracted from a pre-trained BERT model with Earth Mover Distance (EMD) from Rubner et al. (2000) to quantify similarities and dissimilarities between the candidate and the reference summaries.

3.3.2 Datasets

In the following, we describe the datasets used in our experiments.

3.3.2.1 Index datasets

We built various indexes using four corpora, and we indexed different numbers of documents in order to evaluate the robustness of our system.

- **AQUAINT-2** is a News corpus containing 825,148 documents taken from "Agence France Presse" (afp), "Associated Press" (apw), "Xinhua News Agency" (xin), "Central News Agency" (cna), "New York Times" (nyt) and "Los Angeles Times" (ltw). Table 3.2 describes the sources and the number of files for each set of news.

Source	Description	Number of articles
afp_en	Agence France Presse	270,081
apw_eng	Associated Press	187,234
cna_eng	Central News Agency (Taiwan) English Service	14,960
ltw_eng	Los Angeles Times - Washington Post News Service	59,282
nyt_eng	New York Times	152,082
xin_eng	Xinhua News Agency (Beijing) English Service	141,509

Table 3.2: Description of the AQUAINT-2 corpus (as at December 19, 2008)

For experiments, we vary the number of documents D={10000, 15000, 30000, 60000, 179520, 825148}. All indexes are balanced (we take the same number of documents from each subset of the corpus), except for the last one, where we index all documents from AQUAINT-2. We select the files randomly (only one draw was made).

- **Wikipedia** is a free online encyclopedia that contains 1,778,742 documents. Wikipedia contains varied information from many sources and is useful to assess the performance of wikiSERA in the general domain. For experiments, we vary the number of documents D={10000, 15000, 30000, 1778742}. Note that the last number corresponds to the full size of the dataset. Documents are also selected randomly here using only one draw. Finally, since the number of tokens in the evaluated summaries is 100, we select files which contain at least 400 tokens in order to get Compression Radio $CR = (length_{summary})/(length_{text}) <=$ 0.25. The closer the CR score is from zero, the better the summary is Mitkov (2004).

3.3.2.2 Queries datasets

Automatic summaries from the news datasets TAC 2008, TAC 2009, and CN-NDM (Bhandari et al., 2020) (Section 3.3.2) are used as queries. The two TAC datasets are a sub-set of AQUAINT dataset (Graff, 2002).

- **TAC 2008** contains two sets: A and B, where the set B is the updated version of set A. Each set contains 48 topics. Each topic includes 10 documents, where humans provide 4 reference summaries for each topic. The candidate summaries are proposed by 58 participants, where each participant provides one automatic candidate summary per topic. In total, there are 960 documents, 5568 candidate summaries, and 384 reference summaries. For experiments, We index 960 documents, which is the total number of documents in this dataset.

- **TAC 2009** also contains two sets. Each set contains 44 topics. Each topic includes 10 documents, where humans provide 4 reference summaries for each topic. The candidate summaries are proposed by 55 participants, where each participant provides one automatic candidate summary per topic. In total, there are 880 documents, 4840 candidate summaries, and 352 reference summaries. For experiments, We index 880 documents, which is the total number of documents in this dataset.

- **CNN Daily Mail** (Bhandari et al., 2020). This news-based database is of great interest to us because it has candidate summaries obtained from both extractive and abstractive systems. This will help us to check the robustness of wikiSERA to evaluate extractive and abstractive approaches. The used CNNDM dataset consists of 100 reference summaries, having each 25 candidate summaries generated by 11 extractive systems and 14 abstractive systems. We list the extractive and abstractive systems used in Bhandari et al. (2020) to get the 25 candidate summaries for each reference summary. We used these results to compare wikiSERA with the other evaluation measures. The systems are the following:
  - **Extractive systems**: REFRESH (Narayan et al., 2018b), NeuSum (Zhou et al., 2018), BanditSum (Dong et al., 2018), Latent (Zhang et al., 2018), CNN-LSTM-BiClassifier (Kedzie et al., 2018), HIBERT (Zhang et al., 2019b), Sum-PreTr-Enc (Liu and Lapata, 2019a), Transformer-BiClassifier (Zhong et al., 2019), Transformer-Pointer (Zhong et al., 2019), HETER-SUMGRAPH (Wang et al., 2020), MatchSum (Zhong et al., 2020).
  - **Abstractive systems**: GTTP (See et al., 2017), bottom-up (Gehrmann et al., 2018), fastAbsRL (Chen and Bansal, 2018), fastAbsRL-rank (Chen and Bansal, 2018), unilm-v1 (Dong et al., 2019), unilm-v2 (Dong et al., 2019), twoStageRL (Zhang et al., 2019a), pre-summAbs (Liu and Lapata, 2019a), preSummAbs-ext (Liu and Lapata, 2019a), T5 (Raffel et al., 2020), BART (Lewis et al., 2019), SemSim-Summ (Yoon et al., 2020).

3.3.3 Evaluation metric

To compare wikiSERA with other state-of-the-art methods, we use the correlation between the scores provided by each automatic method and the scores provided by manual evaluation metrics.

The manual evaluation metrics used here are:

- **Pyramid** - (Nenkova and Passonneau, 2004) is a manual evaluation measure that exploits the content distribution of human summaries. Pyramid is based on SCUs (Summarization Content Units). A content unit is a set of sub-sentences, at most a clause, that expresses the same semantic content. The Pyramid approach attributes a weight for each SCU depending on its frequency in the summary corpus. Afterward, a pyramid is created where the most frequent SCUs are at the top of the Pyramid, and the less frequent ones are at its bottom. The number of levels in the Pyramid depends on the content units' weight, and the summary's score is computed by dividing the sum of the SCUs weights by an optimal weights sum with the same number of SCUs.

- **LitePyramid** (Shapira et al., 2019) - is a crowdsource-based lightweight version of Pyramid that relies on statistical sampling instead of exhaustive SCU extraction and testing. LitePyramid shows a high and stable correlation with the standard Pyramid method, compared to Responsiveness.

- **Responsiveness** - another manual method suited for Question-Answering problems. It incorporates aspects of linguistic quality to assign a score that measures the quality of a summary.

Correlation metrics that we used are defined as follows:

- **Pearson correlation** (Benesty et al., 2009). It measures the linear relationship between two datasets while assuming them to be normally distributed. Pearson correlation varies between -1 and 1. A zero value means there exists no correlation between the two datasets. A positive correlation means that when x increases, y also does. A negative correlation means that when x increases, y decreases. The formula is given by:

$$r_p = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.1}$$

  where n is the number of samples, x_i is a sample, and $\bar{x}$ is the mean of the x samples.

- **Spearman correlation** (Kokoska and Zwillinger, 2000). It is a non-parametric rank-order correlation that measures the monotonicity of the relationship between two datasets. Spearman correlation does not assume that the two distributions are normally distributed. Correlations of -1 or $+1$ imply an exact monotonic relationship. The formula is given by:

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2 - 1)} \tag{3.2}$$

  where $d_i = x_i - y_i$

- **Kendall tau-b correlation** (Kendall, 1945). It measures the correspondence between two rankings. A value of 1 means a strong agreement, while a value of -1 means a strong disagreement. This version of Kendall's correlation can account for ties contrarily to the first version of this correlation. The formula is given by

$$r_k = \frac{P - Q}{\sqrt{(P+Q+T)(P+Q+U)}} \tag{3.3}$$

where P is the number of concordant pairs, Q is the number of discordant pairs, T is the number of ties that are only in x, and U is the number of ties are only in y. If a tie occurs for the same pair in both x and y, it is not added to either T or U.

3.3.4 Implementation details

SERA and wikiSERA were implemented in Python[1]. As mentioned before, there are four types of queries: raw text, Noun Phrases (NP), KeyWords (KW), and wikiSERA's (nouns + verbs + adjectives). For all kinds of queries, the text is cleaned by removing special characters (such as & and $\geqslant$), numbers, and stop words defined in nltk (Loper and Bird, 2002). The text is converted to a lowercase afterward.

To extract nouns, verbs, and adjectives from the text, we use nltk's Part-Of-Speech Tagger. To extract text keywords, we use the feature extractor from sklearn (Pedregosa et al., 2011). Specifically, we use $TF - IDF$ method to get uni-gram, bi-gram, and tri-gram keywords. Once the extraction is done, extracted tokens are concatenated together before being passed to the index. Note that following the authors Cabrera-Diego and Torres-Moreno (2018), we lowercase the documents and delete numbers and stop-words from them before running SummTriver. Similarly, FRESA applies filtering and stemming on the documents, while ROUGE does not apply preprocessing since its correlations were not affected by stemming or removal of stopwords (Lin, 2004).

To search the queries in the index documents, we used Whoosh (Chaput, 2007-2012), a flexible and pure python search engine framework. We used the default Okapi BM25F ranking function from Whoosh. Equally, we used the authors' public implementations to run ROUGE, SummTriver, and FRESA.

[1] Our system is available at `https://github.com/JessicaLopezEspejel/wikiSERA/`

Following Cabrera-Diego and Torres-Moreno (2018), we evaluate SummTriver using a total of 1800 summaries, where 900 are taken from each of the two sets of TAC 2008 and TAC 2009. Also, we follow the authors and test different numbers of summaries n={2, 5, 10, 15, 30} in the distribution P, since the size of this distribution heavily affects the performance of the approach. FRESA was designed initially for mono-document evaluation. Therefore, the main limitation of our comparison with this system is that we concatenated all the articles of the same topic to be able to run it on TAC 2008 and TAC 2009.

For the sake of comparability, scores are averaged for each participant before computing the correlations with Pyramid and Responsiveness.

To evaluate ROUGE, BERTScore, MoverScore, and JS-2 on the CNNDM dataset, we directly used the scores provided by Bhandari et al. (2020) and computed their correlations with LitePyramid human evaluation. Based on AQUAINT-2 and Wikipedia results, we use an index size D=10000 to run SERA and wikiSERA.

Note that all evaluation methods used here were run on a Central Processing Unit (CPU). It is essential to highlight that the complexity of SERA and wikiSERA is relative to the size of the index, where the biggest the index, the longest the execution time. However, they are fortunately still functional in the absence of a Graphical Processing Unit (GPU). For example, when running wikiSERA on a CPU, it takes 40 minutes, 1 hour, 1 hour, and 30 minutes with index sizes of 10000, 15000, and 30000, respectively. The largest index size is the one from Wikipedia, including 1778742 documents. Here, wikiSERA takes around 8 hours of execution-only. More recent evaluation methods such as SSAS (Vadapalli et al., 2017) and BERTScore (Zhang et al., 2020b) are costly and should be run on a GPU. For example, SSAS needs to train multiple deep semantic models which are used to compute various features (Hermann et al., 2015, Nallapati et al., 2016).

3.4 Results

3.4.1 Correlation on the TAC 2008 dataset

Table 3.4-*left* shows the correlation coefficients on the TAC 2008 dataset of ROUGE, SERA, wikiSERA, SummTriver, and FRESA with two manual evaluation approaches: Pyramid and Responsiveness. We computed correlation coefficients with Pyramid using both four and three manual reference summaries, and the results are slightly different in favor of the latter. For this reason, we present in Table 3.4-*left* correlation scores using three reference summaries for Pyramid, while we use the four reference summaries for SERA and wikiSERA. Detailed results are in Appendix A.

We experimented with SERA and wikiSERA metrics by building various indexes from AQUAINT-2 and Wikipedia corpora. Results show that, in AQUAINT-2, the wikiSERA-5 method outperforms the SERA method when indexing 15000 documents. It outperforms SERA by approximately 0.2 points for Pyramid, and by almost 0.3 points for Responsiveness. Scores in terms of Pearson and Spearman of wikiSERA with AQUAINT-2 are fairly close to ROUGE, with the latter providing the highest correlations with both Pyramid and Responsiveness. Besides, SERA and wikiSERA outperform the scores achieved by SummTriver and FRESA with both Pyramid and Responsiveness in terms of the three types of correlation tested here.

When we index documents from Wikipedia corpus using D=30000, we achieve the highest correlation with SERA-DIS-NP-10, while the best wikiSERA variant is wikiSERA-10. Once again, SERA outperforms the scores of SummTriver and FRESA with both Pyramid and Responsiveness. In wikiSERA, this approach achieves higher correlation coefficients in terms of Pearson and Spearman measures with Pyramid and higher correlation in terms of Pearson measure with Responsiveness. The SummTriver system provides higher scores than wikiSERA against Pyramid when we use 900 summaries per corpus. Finally, the FRESA baseline achieves the lowest correlation scores with manual methods. Here, the evaluation was done using summaries from all participants. Also important, we concatenated each topic's documents in order to be able to run this approach. As mentioned before, this is the main limitation of our comparison with FRESA since it was designed for mono-document summary evaluation.

Figure 3.2 shows the Pearson correlation coefficients with both Pyramid and Responsiveness when indexing different numbers of documents from the AQUAINT-2 dataset. We observe that the highest scores are obtained with D={10000, 15000, 30000, 60000} documents in the index dataset. Consequently, it is better to use a limited set of documents instead of all the corpus. Moreover, Figure 3.3 shows the Pearson correlation coefficients with both Pyramid and Responsiveness when indexing different numbers of documents from Wikipedia. Once again, the lowest scores appear when we index all the documents from the dataset.

Because the best results with AQUAINT-2 were obtained using small index sizes, we varied the number of index documents in Wikipedia between D={10000, 15000, 30000} and all documents from the corpus. Obtained results confirm that the best

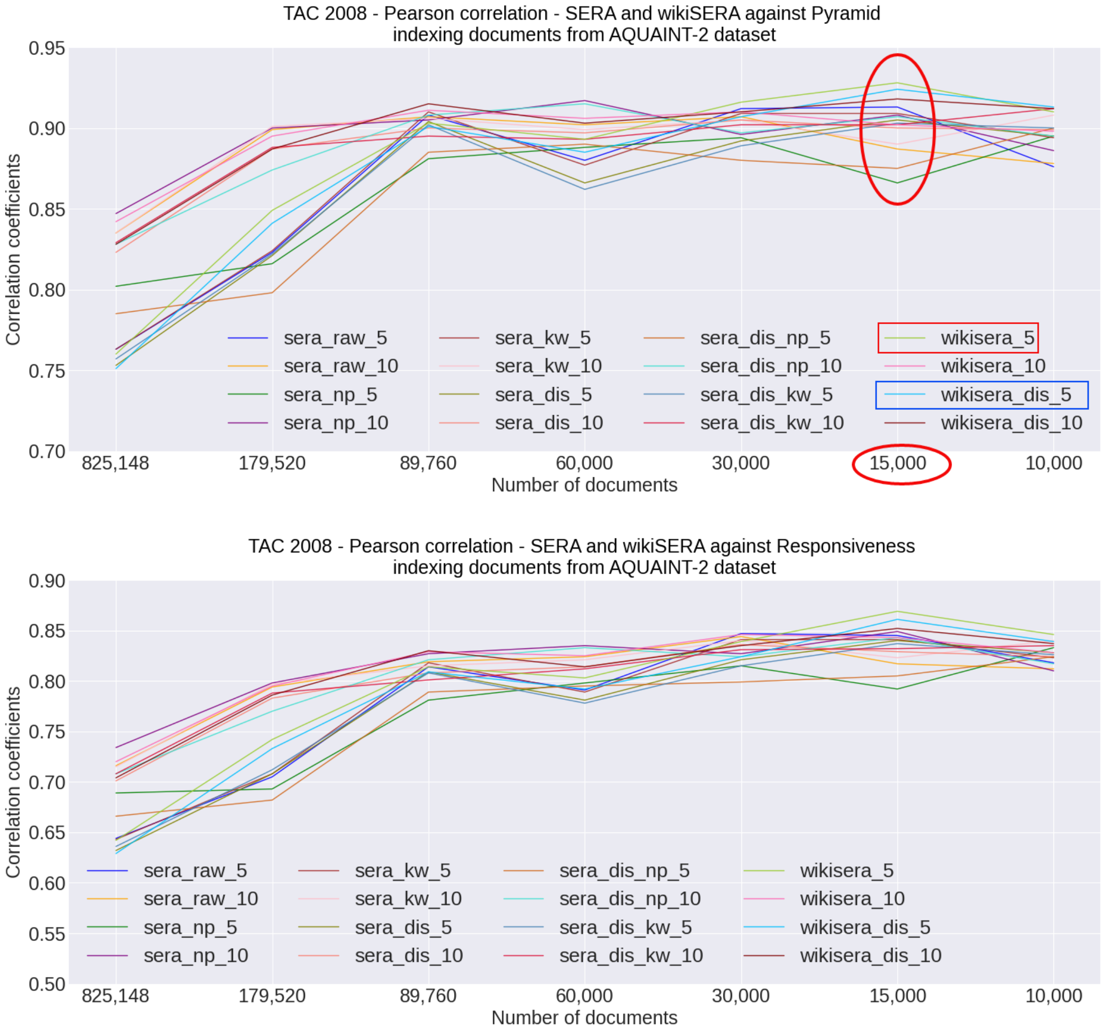


Figure 3.2: Pearson correlation coefficients using TAC 2008 dataset as queries and AQUAINT-2 documents as an index

number of index documents varies between 10000 and 30000.

3.4.2 Correlation on the TAC 2009 dataset

Table 3.4-*right* shows correlation coefficients of ROUGE, SERA, wikiSERA, SummTriver, and FRESA against Pyramid and Responsiveness using TAC 2009 corpus as queries, while using AQUAINT-2 and Wikipedia as index datasets. Correlation coefficients were obtained using the four reference summaries scores for all automatic approaches. In contrast, only three of them were used for Pyramid since they provide slightly better results than the four manual scores. Detailed results are in Appendix A.

ROUGE presents the highest scores against SERA and wikiSERA when we index

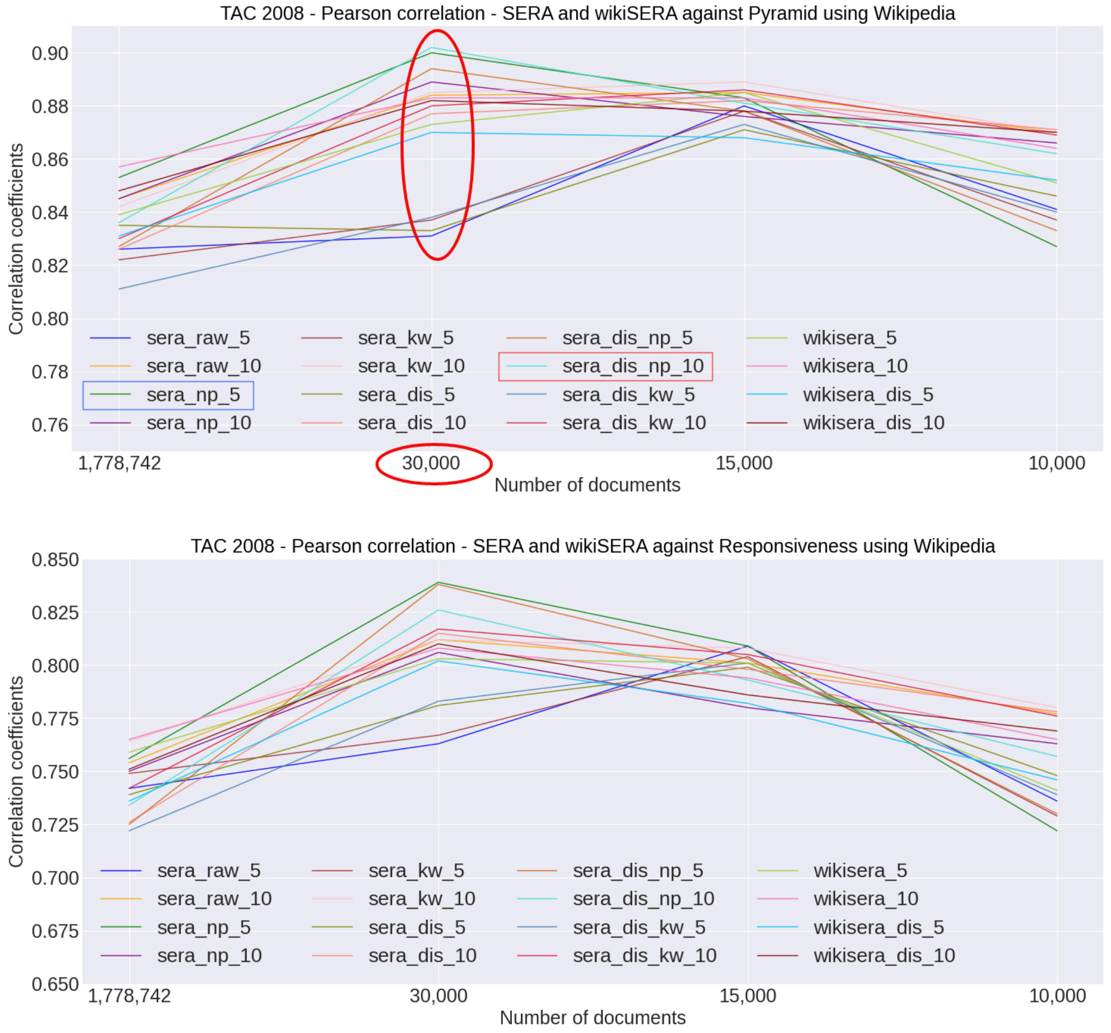


Figure 3.3: Pearson correlation coefficients using TAC 2008 dataset as queries and Wikipedia documents as an index

documents from AQUAINT-2. However, the best results from AQUAINT-2 were obtained when indexing 179520 documents. In addition, the wikiSERA method achieves better scores than SERA. Actually, wikiSERA-DIS-5 and wikiSERA-5 obtain the highest scores compared to the Pyramid method, and the Responsiveness manual evaluation methods, respectively.

SERA and wikiSERA outperform, in terms of Pearson correlation, ROUGE against the Pyramid manual method when indexing 10,000 documents from Wikipedia. The best scores are achieved by SERA-DIS-10 in terms of Pearson and Kendall correlations, while SERA-DIS-KW-10 behaves better when it comes to the Spearman correlation. On the other hand, SERA and wikiSERA also outperform the ROUGE method with Responsiveness in terms of Pearson correlation. For SERA methods, we get the best scores with SERA-NP-10 in terms of the three types of correlation methods tested here. However, in wikiSERA, we get the highest Pearson correlation with wikiSERA-10, while the highest scores are obtained with wikiSERA-DIS-10 in terms of Spearman and Kendall correlations.

Results reported in Table 3.4-*right* were obtained using the four manual annotators in Pyramid contrarily to the experiments with TAC 2008. The highest Pearson correlation achieved is **0.959**. This score is obtained with SERA-DIS-10 and wikiSERA-DIS-10 using the average of three annotator scores.

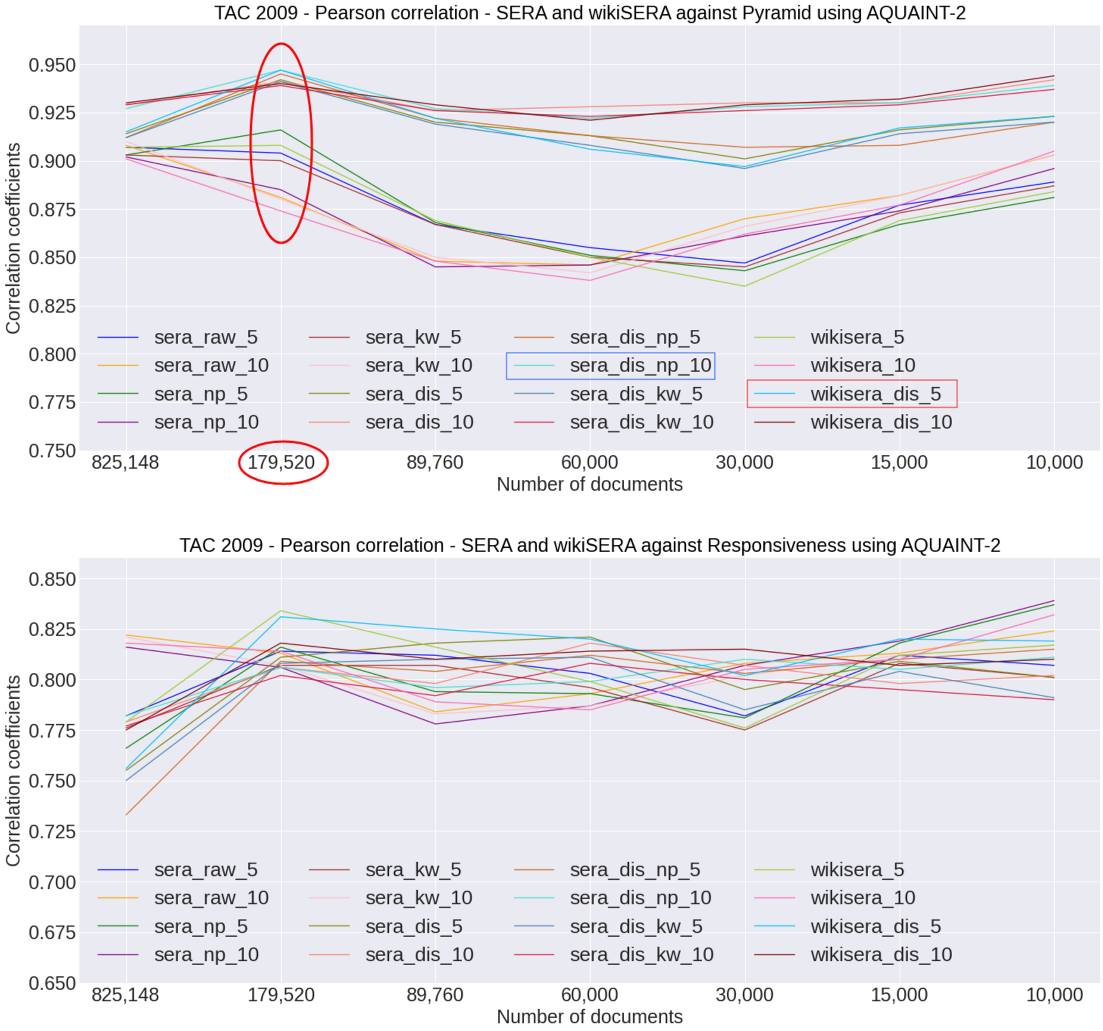


Figure 3.4: Pearson correlation coefficients using TAC 2009 dataset as queries and AQUAINT-2 documents as an index

Figure 3.4 and 3.5 show the Pearson correlation coefficients with both Pyramid and Responsiveness when indexing a different number of documents from AQUAINT-2 and Wikipedia datasets. This study shows that SERA and wikiSERA outperform in some cases the scores obtained with ROUGE variants, and they are closer to manual metrics.

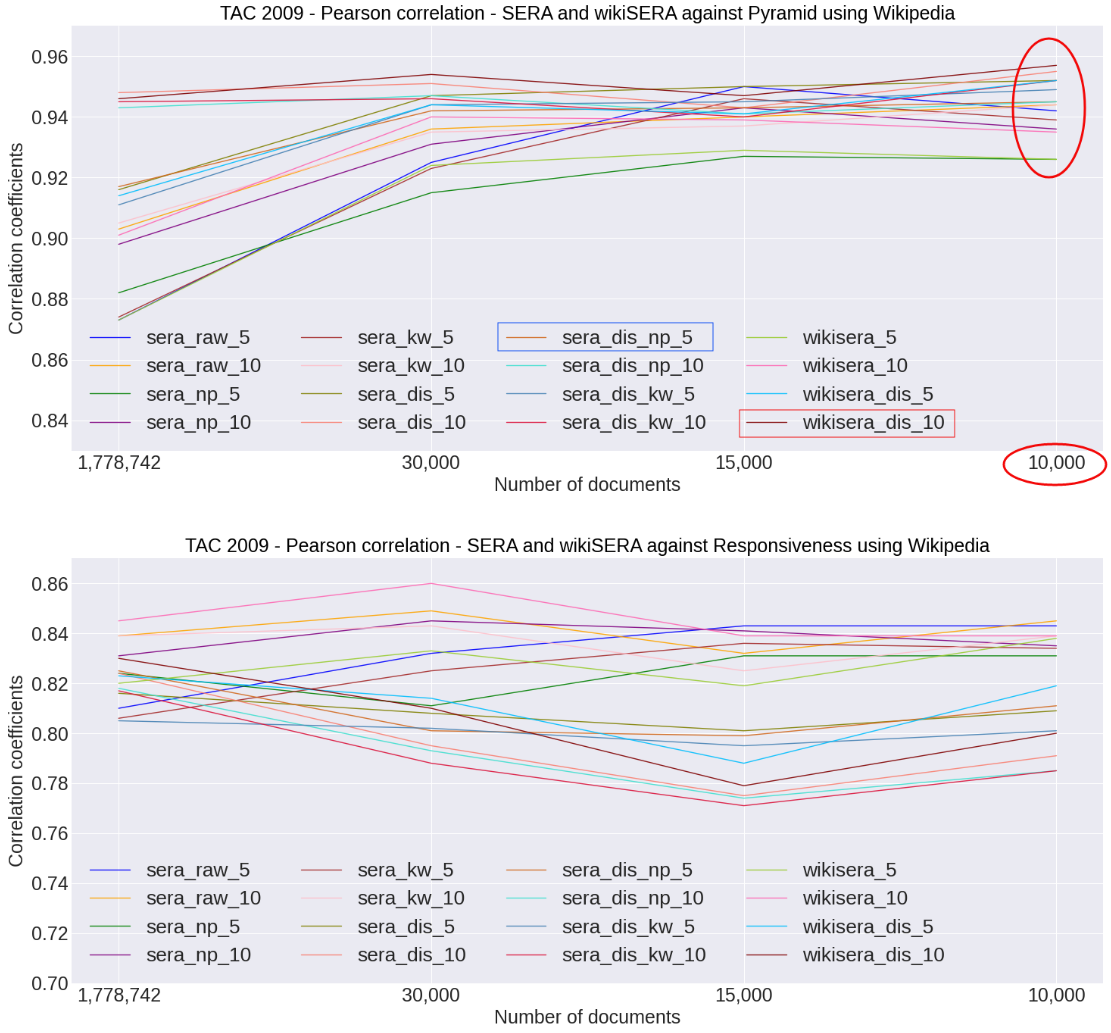


Figure 3.5: Pearson correlation coefficients using TAC 2009 dataset as queries and Wikipedia documents as an index

3.4.3 Correlation on the CNNDM dataset

Table 3.3-$left$ shows the correlation coefficients in terms of Pearson, Spearman, and Kendall, of SERA, wikiSERA, ROUGE, BERTScore, JS-2, and MoverScore with LitePyramid using both extractive and abstractive summaries from CNNDM (Bhandari et al., 2020) dataset. To analyze the performance of different systems on each type of summarization, we present in Table 3.3-*middle* and Table 3.3-*right* the correlations using only the extractive summaries and only the abstractive summaries, respectively.

We first analyze Table 3.3-*left* since it provides an overview of the global behavior of each system. Results show that the highest correlations of ROUGE are obtained with ROUGE-2-Recall. Globally, the highest correlations in ROUGE are obtained with the recall metric (ROUGE-R), followed by the ROUGE-F measure, and finally by ROUGE-P. The second highest correlations are obtained with wikiSERA-10, while the third highest correlations come from SERA. Although SERA-KW-10 has the best score in Spearman and Kendall, all the SERA variants present very similar scores. Behind the SERA method, BERScore and JS-2 measures present very similar scores. Meanwhile, MoverScore shows the lowest correlations.

According to Table 3.3-*middle*, the highest correlation when we evaluate only extractive summaries is obtained with ROUGE-2-Recall in terms of Pearson, and ROUGE-1-Recall in terms of Spearman and Kendall. This finding is not surprising since extractive approaches directly copy-paste sentences from the source document to the summary, while humans write more abstractive reference summaries, leading to a high matching of uni-grams and bi-grams only between candidate and reference summaries.

The second metric achieving the highest correlation is SERA-DIS-NP-5. Unlike the correlations obtained with both extractive and abstractive summaries (Table 3.3-*left*), the correlations obtained using only extractive summaries vary considerably for SERA. For instance, the difference in terms of Pearson score between SERA-DIS-NP-5 and SERA-DIS-10 is 0.502. Behind SERA comes wikiSERA, where the highest scores are obtained with wikiSERA-5 in Pearson, Kendall, and wikiSERA-10 in Spearman correlation. Once again, wikiSERA overcomes JS-2, BERTScore, and MoverScore.

Finally, Table 3.3-*right* shows the correlations between tested approaches and human evaluation using only abstractive summaries. Once again, the highest results in ROUGE come from the Recall metric. Interestingly, ROUGE-2-R keeps the highest correlations in comparison with the other tested metrics. Note that ROUGE-2-R presents the same correlations with ROUGE-L-F in terms of Spearman and Kendall. Based on these observations, we conclude that abstractive approaches can produce larger common phrases between candidate and reference summaries than extractive ones, where we find more uni-grams matching between them. Moreover, wikiSERA outperforms once again the BERTScore, MoverScore, and JS-2 methods. It is noteworthy that scores in this table are considerably higher than those in

		Both extractive and abstractive			Extractive			Abstractive		
		Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
ROUGE	ROUGE-1-F	0.600	0.468	0.358	0.111	-0.018	0.055	0.879	0.938	0.824
	ROUGE-1-P	-0.175	-0.212	-0.117	-0.179	-0.082	-0.055	0.308	0.367	0.275
	ROUGE-1-R	0.914	0.922	0.773	0.703	**0.691**	**0.564**	0.913	0.705	0.538
	ROUGE-2-F	0.648	0.452	0.311	0.190	-0.091	-0.055	0.876	0.938	0.802
	ROUGE-2-P	0.099	0.050	0.023	-0.044	-0.082	-0.055	0.536	0.688	0.473
	ROUGE-2-R	**0.962**	**0.958**	**0.860**	**0.739**	0.618	0.491	**0.983**	**0.947**	**0.868**
	ROUGE-L-F	0.526	0.368	0.278	0.091	-0.036	-0.018	0.707	**0.947**	**0.868**
	ROUGE-L-P	-0.045	-0.148	-0.070	-0.188	-0.109	-0.055	0.372	0.367	0.319
	ROUGE-L-R	0.871	-0.914	0.759	0.699	0.555	0.418	0.832	0.727	0.582
BERTS	BERTScore-1-F	0.385	0.374	0.258	-0.068	-0.073	0.055	0.631	0.635	0.495
	BERTScore-1-P	-0.021	0.093	0.064	-0.290	-0.136	**-0.127**	0.371	0.508	0.341
	BERTScore-1-R	**0.768**	**0.738**	**0.552**	**0.333**	**0.191**	**0.127**	**0.824**	**0.719**	**0.626**
MS	MoverScore	**0.443**	**0.367**	**0.284**	**0.012**	**-0.009**	**-0.018**	**0.858**	**0.956**	**0.868**
JS	JS-2	**0.780**	**0.665**	**0.512**	**0.129**	**-0.064**	**0.018**	**0.902**	**0.947**	**0.824**
SERA and wikiSERA with Wikipedia	SERA-5	0.773	0.710	0.508	0.143	-0.082	-0.127	0.866	0.759	0.522
	SERA-10	0.858	**0.789**	0.616	0.032	-0.036	-0.091	0.925	0.868	0.736
	SERA-NP-5	0.690	0.639	0.452	0.483	0.421	0.337	0.665	0.700	0.508
	SERA-NP-10	0.743	0.705	0.502	0.451	0.393	0.278	0.761	0.823	0.619
	SERA-KW-5	0.784	0.711	0.508	0.165	-0.141	-0.147	0.871	0.808	0.589
	SERA-KW-10	**0.864**	0.782	**0.621**	0.050	-0.027	-0.073	**0.941**	0.904	0.773
	SERA-DIS-5	0.748	0.668	0.472	0.421	0.309	0.164	0.870	0.824	0.582
	SERA-DIS-10	0.827	0.781	0.605	0.102	0.091	0.018	0.904	**0.925**	**0.802**
	SERA-DIS-NP-5	0.599	0.554	0.391	**0.604**	**0.618**	**0.491**	0.622	0.591	0.385
	SERA-DIS-NP-10	0.671	0.568	0.393	0.593	0.573	0.455	0.718	0.744	0.486
	SERA-DIS-KW-5	0.758	0.657	0.465	0.438	0.309	0.164	0.871	0.815	0.560
	SERA-DIS-KW-10	0.828	0.752	0.565	0.143	0.164	0.091	0.910	0.921	0.780
	wikiSERA-5	0.623	0.527	0.387	**-0.586**	-0.536	**-0.418**	0.803	0.667	0.522
	wikiSERA-10	**0.880**	**0.872**	**0.719**	0.479	**0.548**	0.389	**0.903**	0.710	**0.560**
	wikiSERA-DIS-5	0.566	0.469	0.315	-0.339	-0.464	-0.345	0.763	0.653	0.473
	wikiSERA-DIS-10	0.817	0.788	0.605	0.409	0.345	0.200	0.885	**0.727**	**0.560**

Table 3.3: Correlation coefficients on CNNDM dataset, in terms of Pearson, Spearman and Kendall, of multiple automatic evaluation methods with LitePyramid using both extractive and abstractive summaries (left), only extractive summaries (middle), and only abstractive summaries (right). Best results of each method are in bold.

Table 3.3-*middle* (only extractive summaries). Unlike the results from Tables 3.3-*left* and 3.3-*right*, the lowest correlation scores are obtained with BERTScore-1-R when we use only abstractive summaries.

		TAC 2008						TAC 2009					
		Pyramid			Responsiveness			Pyramid			Responsiveness		
		Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
ROUGE	ROUGE-1-F	0.908	0.941	0.787	0.853	0.883	0.702	**0.951**	0.915	0.788	**0.835**	0.793	0.622
	ROUGE-1-P	0.730	0.841	0.643	0.698	0.803	0.626	0.923	0.845	0.678	0.791	0.789	0.630
	ROUGE-1-R	0.911	0.935	0.774	0.851	0.858	0.665	0.926	0.892	0.748	0.814	0.764	0.591
	ROUGE-2-F	0.940	0.965	0.843	0.892	0.915	0.746	0.930	0.955	0.839	0.740	0.831	0.664
	ROUGE-2-P	0.911	0.942	0.788	0.873	0.901	0.730	0.906	0.937	0.796	0.716	0.829	0.658
	ROUGE-2-R	**0.946**	**0.967**	**0.851**	0.894	0.918	0.755	0.937	0.952	0.841	0.746	0.820	0.654
	ROUGE-3-F	0.941	0.951	0.810	**0.915**	**0.924**	**0.767**	0.842	**0.964**	0.841	0.622	**0.852**	**0.675**
	ROUGE-3-P	0.926	0.934	0.783	0.909	0.918	0.766	0.828	0.940	0.800	0.610	0.839	0.656
	ROUGE-3-R	0.945	0.951	0.811	0.914	0.922	0.763	0.848	**0.964**	**0.845**	0.627	0.845	0.673
	ROUGE-L-F	0.878	0.925	0.756	0.823	0.868	0.689	0.865	0.604	0.461	0.649	0.414	0.294
	ROUGE-L-P	0.711	0.823	0.632	0.679	0.794	0.611	0.801	0.546	0.406	0.573	0.360	0.255
	ROUGE-L-R	0.882	0.927	0.762	0.823	0.856	0.661	0.875	0.622	0.474	0.663	0.414	0.298
	ROUGE-W-1.2-F	0.901	0.940	0.782	0.848	0.878	0.701	0.882	0.654	0.512	0.651	0.462	0.341
	ROUGE-W-1.2-P	0.712	0.822	0.631	0.688	0.794	0.620	0.798	0.514	0.393	0.558	0.337	0.237
	ROUGE-W-1.2-R	0.897	0.940	0.785	0.841	0.871	0.684	0.889	0.671	0.529	0.659	0.469	0.340
	ROUGE-SU4-F	0.917	0.949	0.805	0.870	0.904	0.728	0.934	0.940	0.818	0.747	0.808	0.639
	ROUGE-SU4-P	0.839	0.910	0.728	0.805	0.869	0.689	0.893	0.910	0.761	0.702	0.804	0.638
	ROUGE-SU4-R	0.927	0.950	0.800	0.874	0.908	0.736	0.942	0.924	0.787	0.756	0.789	0.619
SERA and wikiSERA with AQUAINT-2	SERA-5	**0.913**	**0.908**	0.732	0.845	0.821	0.624	0.904	0.818	0.656	0.814	0.664	0.502
	SERA-10	0.887	0.871	0.693	0.817	0.766	0.572	0.881	0.817	0.651	0.813	0.675	0.513
	SERA-NP-5	0.866	0.863	0.681	0.792	0.770	0.569	0.916	0.831	0.670	**0.816**	0.692	**0.530**
	SERA-NP-10	0.908	0.905	**0.739**	**0.849**	**0.827**	**0.632**	0.885	0.828	0.662	0.806	**0.702**	0.529
	SERA-KW-5	0.909	0.901	0.721	0.841	0.809	0.611	0.900	0.816	0.654	0.807	0.665	0.503
	SERA-KW-10	0.890	0.880	0.705	0.823	0.779	0.579	0.880	0.810	0.646	0.807	0.670	0.511
	SERA-DIS-5	0.905	0.885	0.713	0.840	0.800	0.593	0.942	0.829	0.666	0.811	0.660	0.501
	SERA-DIS-10	0.900	0.888	0.711	0.829	0.797	0.592	0.941	**0.836**	**0.671**	0.806	0.673	0.521
	SERA-DIS-NP-5	0.875	0.864	0.679	0.805	0.774	0.573	0.945	0.831	0.670	0.809	0.687	0.529
	SERA-DIS-NP-10	0.907	0.905	0.735	0.843	0.819	0.616	**0.947**	0.825	0.665	0.806	0.683	0.518
	SERA-DIS-KW-5	0.903	0.885	0.712	0.837	0.801	0.597	0.941	0.822	0.659	0.808	0.653	0.496
	SERA-DIS-KW-10	0.902	0.888	0.709	0.832	0.804	0.601	0.939	0.826	0.658	0.802	0.669	0.515
	wikiSERA-5	**0.928**	**0.924**	**0.760**	**0.869**	**0.854**	**0.663**	0.908	0.835	0.678	**0.834**	**0.697**	**0.530**
	wikiSERA-10	0.902	0.890	0.708	0.843	0.800	0.604	0.874	0.813	0.652	0.814	0.686	0.517
	wikiSERA-DIS-5	0.924	0.910	0.746	0.861	0.836	0.641	**0.947**	**0.836**	**0.688**	0.831	0.684	0.525
	wikiSERA-DIS-10	0.918	0.896	0.724	0.852	0.806	0.610	0.940	0.818	0.657	0.818	0.673	0.514
SERA and wikiSERA with Wikipedia	SERA-5	0.831	0.839	0.673	0.763	0.751	0.560	0.942	0.870	0.717	0.843	0.768	0.592
	SERA-10	0.884	0.900	0.724	0.812	0.798	0.594	0.944	0.892	0.741	**0.845**	**0.784**	**0.607**
	SERA-NP-5	0.900	0.898	0.733	**0.839**	0.819	0.616	0.926	0.863	0.704	0.831	0.749	0.573
	SERA-NP-10	0.890	0.912	0.738	0.806	0.812	0.618	0.936	0.863	0.709	0.835	0.759	0.592
	SERA-KW-5	0.837	0.838	0.667	0.767	0.749	0.552	0.939	0.863	0.701	0.834	0.761	0.588
	SERA-KW-10	0.885	0.906	0.727	0.812	0.806	0.603	0.944	**0.894**	0.738	0.839	0.771	0.588
	SERA-DIS-5	0.833	0.825	0.655	0.781	0.757	0.568	0.952	0.877	0.729	0.809	0.778	0.602
	SERA-DIS-10	0.877	0.887	0.707	0.815	0.790	0.588	**0.955**	0.896	**0.751**	0.791	0.781	0.603
	SERA-DIS-NP-5	0.894	0.884	0.718	0.838	0.809	0.604	0.945	0.842	0.684	0.811	0.733	0.563
	SERA-DIS-NP-10	**0.902**	**0.917**	**0.754**	0.826	**0.820**	**0.626**	0.945	0.845	0.688	0.785	0.746	0.579
	SERA-DIS-KW-5	0.838	0.837	0.667	0.783	0.761	0.567	0.949	0.868	0.713	0.801	0.773	0.596
	SERA-DIS-KW-10	0.881	0.894	0.719	0.817	0.797	0.598	0.952	**0.899**	0.753	0.785	0.782	**0.607**
	wikiSERA-5	0.873	0.865	0.698	0.803	0.774	0.581	0.926	0.854	0.701	0.838	0.737	0.570
	wikiSERA-10	**0.883**	**0.903**	**0.727**	0.808	**0.805**	0.598	0.935	0.870	0.710	**0.839**	0.737	0.571
	wikiSERA-DIS-5	0.870	0.865	0.701	0.802	0.773	0.580	0.952	0.867	**0.717**	0.819	**0.768**	**0.592**
	wikiSERA-DIS-10	0.882	0.899	0.722	**0.810**	0.800	**0.601**	**0.957**	**0.882**	0.710	0.800	0.748	0.577
SummTriver	ST-JS-$\mathcal{T}_m$	**-0.889**	**-0.827**	**-0.643**	**-0.820**	**-0.801**	**-0.608**	**-0.526**	**-0.755**	**-0.623**	**-0.650**	**-0.744**	**-0.587**
	ST-sJS-$\mathcal{T}_m$	-0.885	-0.822	-0.637	-0.822	-0.797	-0.605	-0.511	-0.751	-0.620	-0.636	-0.739	-0.585
	ST-KL-$\mathcal{T}_m$	-0.694	-0.700	-0.510	-0.706	-0.695	-0.504	-0.371	-0.681	-0.558	-0.518	-0.683	-0.550
	ST-JS-$\mathcal{T}_c$	-0.858	-0.805	-0.613	-0.771	-0.777	-0.578	-0.477	-0.718	-0.582	-0.619	-0.710	-0.563
	ST-sJS-$\mathcal{T}_c$	-0.857	-0.805	-0.612	-0.771	-0.777	-0.577	-0.475	-0.717	-0.581	-0.618	-0.709	-0.562
	ST-KL-$\mathcal{T}_c$	-0.216	-0.168	-0.123	0.025	0.134	0.091	-0.138	-0.062	-0.040	-0.014	-0.007	-0.005
FRESA	FRESA-1	-0.487	**-0.638**	**-0.537**	-0.385	-0.498	**-0.371**	-0.610	**-0.650**	**-0.491**	**-0.594**	**-0.565**	**-0.410**
	FRESA-2	0.474	-0.062	-0.064	0.523	0.076	0.034	**-0.630**	0.046	-0.026	-0.385	-0.074	-0.063
	FRESA-3	0.539	0.241	0.162	0.593	0.362	0.250	-0.556	0.055	0.056	-0.298	0.180	-0.147
	FRESA-4	**0.544**	0.257	0.168	**0.596**	**0.416**	0.296	-0.516	0.189	0.142	-0.217	0.363	-0.278

Table 3.4: Correlation coefficients of TAC 2008 (left) and TAC 2009 (right) datasets, in terms of Pearson, Spearman and Kendall, of multiple automatic evaluation methods with Pyramid and Responsiveness. Best results of each method are in bold.

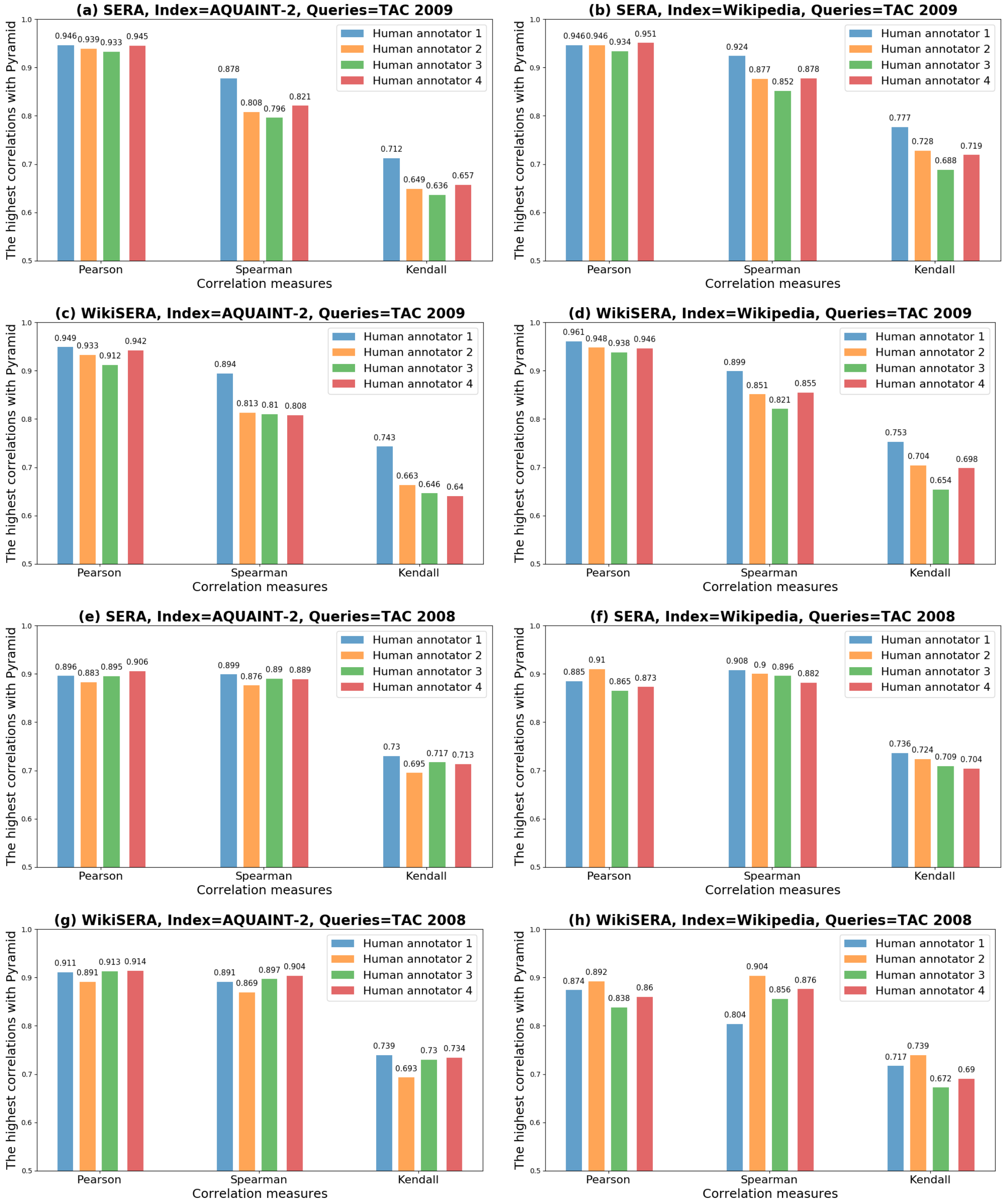
3.4.4 Impact of human annotators on SERA and wikiSERA

In Appendix A, we provide extensive ablation experiments to study the impact of the human annotators in TAC 2008 and TAC 2009 datasets when we evaluate summaries with SERA (the evaluation method proposed by Cohan and Goharian (2016)) and wikiSERA (the evaluation approach proposed in this thesis). In these experiments, we compute the correlation using: (1) each human annotator $\mathcal{A}$ individually ($\mathcal{A}_{i \in [1,4]}$), (2) three human annotators ($\mathcal{A}_1$ & $\mathcal{A}_2$ & $\mathcal{A}_3$), ($\mathcal{A}_1$ & $\mathcal{A}_2$ & $\mathcal{A}_4$), ($\mathcal{A}_2$ & $\mathcal{A}_3$ & $\mathcal{A}_4$), and (3) all human annotators ($\mathcal{A}_1$ & $\mathcal{A}_2$ & $\mathcal{A}_3$ & $\mathcal{A}_4$).

We show experimentally that human annotators affect the performance of automatic evaluation approaches. This finding is intuitive insofar as text summarization is a hard task even for humans. Depending on the expertise of each annotator, quality of manually written summaries varies, leading to biases in automatic evaluation. In Figure 3.6, we show correlations of SERA and wikiSERA with the Pyramid manual evaluation method, while considering the individual score of each annotator $\mathcal{A}_{i \in [1,4]}$. We thus provide the annotator(s) who achieve(s) the highest and lowest correlations in each index-query combination. Results with Responsiveness follow the same trend as with Pyramid.

TAC 2009 query dataset - Figures 3.6-a, 3.6-b, 3.6-c, and 3.6-d provide SERA and wikiSERA correlations with Pyramid using TAC 2009 as a query dataset and AQUAINT-2 and Wikipedia as indexes. Results show that the best human annotator is always $\mathcal{A}_1$ as he provides summaries with the best SERA and wikiSERA correlations in terms of Pearson, Spearman, and Kendall. Alternatively, the human annotator $\mathcal{A}_3$ always gets the lowest correlations in terms of all correlation metrics used.

In Table 3.5, we compare results obtained with the four manual annotators versus those obtained with the best three annotators ($\mathcal{A}_1$ & $\mathcal{A}_2$ & $\mathcal{A}_4$) for TAC 2009. Results show there is a clear gain between using four and three manual annotators, to the favor of the latter for all reported cases. For AQUAINT-2, the gain for SERA is 0.002, 0.01, and 0.016 in terms of Pearson, Spearman, and Kendall, respectively. The gain for wikiSERA is 0.004, 0.005, and 0.023. Alternatively, for Wikipedia, the gain for SERA is 0.004, 0.013, and 0.025 in terms of the three correlation methods, while the gain for wikiSERA is 0.002 and 0.033 in terms of Pearson and Kendall, respectively. We conclude that the automatic evaluation methods that rely on human intervention participate partially in automatic summary evaluation, while the human annotators bias the automatic evaluation based on the quality of their manually written summaries.



(a) SERA, Index=AQUAINT-2, Queries=TAC 2009

(b) SERA, Index=Wikipedia, Queries=TAC 2009

(c) WikiSERA, Index=AQUAINT-2, Queries=TAC 2009

(d) WikiSERA, Index=Wikipedia, Queries=TAC 2009

(e) SERA, Index=AQUAINT-2, Queries=TAC 2008

(f) SERA, Index=Wikipedia, Queries=TAC 2008

(g) WikiSERA, Index=AQUAINT-2, Queries=TAC 2008

(h) WikiSERA, Index=Wikipedia, Queries=TAC 2008

Figure 3.6: Impact of human annotators on the performance of SERA and wikiSERA on TAC 2008 and TAC 2009 datasets

TAC 2008 query dataset - Figures 3.6-e, 3.6-f, 3.6-g, and 3.6-h provide SERA and wikiSERA correlations with Pyramid using TAC 2008 as a query dataset and AQUAINT-2 and Wikipedia as indexes. Like in TAC 2009, human annotators affect automatic evaluation with SERA and wikiSERA. In contrast, it is hard to retrieve a pattern from TAC 2008 figures, as the best human annotator changes from one case to another. For instance, $\mathcal{A}_1$ is the best human annotator for SERA with both AQUAINT-2 and Wikipedia corpora in terms of Spearman and Kendall. However, in terms of Pearson correlation, the best annotator is $\mathcal{A}_4$ for AQUAINT-2 and $\mathcal{A}_2$ for Wikipedia. Alternatively, the best annotator for wikiSERA is always $\mathcal{A}_2$ for Wikipedia, while the same annotator provides the lowest results with AQUAINT-2.

	AQUAINT-2				Wikipedia			
	SERA-DIS-NP-10		wikiSERA-DIS-5		SERA-DIS-10		wikiSERA-DIS-10	
Annotators	4	3	4	3	4	3	4	3
Pearson	0.947	**0.949**	0.947	**0.951**	0.955	**0.959**	0.957	**0.959**
Spearman	0.825	**0.835**	0.836	**0.841**	0.896	**0.909**	**0.882**	**0.882**
Kendall	0.665	**0.681**	0.668	**0.691**	0.751	**0.776**	0.710	**0.743**

Table 3.5: Impact of human annotators on the evaluation with SERA and wikiSERA using TAC 2009.

3.5 Discussion

The intuition behind SERA (Cohan and Goharian, 2016) is that a summary context is represented by its most related articles. Thus, two summaries related to the same documents are semantically related, even if they are lexically different. Consequently, SERA is fairer to evaluate abstractive summaries contrarily to the lexical-based ROUGE. However, SERA suffers from a series of limitations: (1) the code is not open-source, (2) no information was provided concerning the subset of PubMed used as an index, and (3) PubMed is specialized in the biomedical domain only. The first two drawbacks make SERA unusable by the community, while the third restricts its usage to the biomedical domain. We build on SERA merits and limitations to propose wikiSERA, an open-source version of SERA that evaluates summaries from the general domain. Novelties of wikiSERA are the index pool and query reformulation adapted to the evaluation of summaries from the general domain. We use Wikipedia as an index, and this dataset is public. It is thus possible to use wikiSERA to evaluate user summaries. Equally important, we make the code open-source to allow researchers to reproduce our results and improve further automatic summary evaluation.

The performance of SERA depends on the number of index documents and the domain to which they belong. In order to know how many documents are necessary to get the highest correlation scores and evaluate the robustness of our system, we indexed different numbers of documents belonging to different corpora. Depending on the use case, we can select, for instance, 10000 documents in order to make a compromise between the time needed for evaluation and the desired scores we want to achieve.

According to Kieuvongngam et al. (2020), nouns in generated summaries represent more accurately the information conveyed by the original abstracts than other POS tags. With this study, we can explain why Cohan and Goharian (2016) got a better correlation than ROUGE when they defined queries using noun phrases. However, migrating SERA to the general domain implies redefining the most relevant POS Tags in the texts used for experimentation. We lead a POS-tag analysis to know the distribution of nouns, verbs, adjectives, prepositions, and other POS Tags on three corpora: PubMed (Cohan and Goharian, 2016), AQUAINT-2 (Consortium, 2008), and our Wikipedia corpus. Our study on three different corpora belonging to different domains shows that the percentages of verbs and adjectives are higher in AQUAINT-2 (news) and Wikipedia (general domain) than PubMed dataset. Therefore, we propose a novel method based on query reformulation: wikiSERA. Our approach extracts from the query nouns, verbs, and adjectives.

wikiSERA shows a better behavior for TAC 2009 than for TAC 2008 (Table 3.4). Indeed, wikiSERA-DIS-10, the best variant of wikiSERA for TAC 2009, achieves better results than ROUGE with Pearson correlation when using Wikipedia as an index. This finding proves the effectiveness of wikiSERA to evaluate summaries from the general domain. Equally, wikiSERA reduces the gap between SERA and ROUGE in most of the other cases.

Furthermore, we extensively study the performance variation of both SERA and wikiSERA when computing the correlation using: (1) the score of one human annotator ($\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, or $\mathcal{A}_4$), (2) the average score of three human annotators ($\mathcal{A}_1$ & $\mathcal{A}_2$ & $\mathcal{A}_3$), ($\mathcal{A}_1$ & $\mathcal{A}_2$ & $\mathcal{A}_4$), ($\mathcal{A}_2$ & $\mathcal{A}_3$ & $\mathcal{A}_4$), and (3) the average score of all human annotators. Experiments show that some annotators get, in most cases, a high correlation with Pyramid and Responsiveness. For example, in TAC 2009 dataset, the first annotator ($\mathcal{A}_1$) achieves the highest correlations against Pyramid and Responsiveness. Inversely, the third manual annotator gets the lowest correlations when we analyze his scores individually.

The experiments we led helped us to define the corpus properties and to study the impact of human annotators and the index size on the performance of SERA and wikiSERA. The POS Tags study was also relevant to propose a fair query redefinition to evaluate summaries belonging to a general domain.

SummTriver achieves reasonably good results even without using any human reference. This system is beneficial when human summaries are costly or hard to find. However, when such references are available, SummTriver does not take advantage of them, leading its correlation to be low compared to human-reference-based evaluation approaches such as ROUGE and SERA. Note that the best results of SummTriver were obtained with $n = 30$ summaries in the distribution P for Pyramid and with $n = 10$ summaries for Responsiveness. More results are in Appendix A.

FRESA achieves the lowest correlation scores with manual methods. Its performance drops approximately from 0.1 to 0.3 points compared to the lowest results obtained by SERA. This is mainly because FRESA is based only on the divergence between the evaluated summary and its source documents, without including any comparison with summaries generated by other participants, as Summtriver does. Thus, FRESA is barely correlated with manual evaluation in many cases where the correlation gets close to zero (for instance, FRESA-2 with TAC 2009 using Kendall correlation). Note that SummTriver and FRESA have mostly negative correlations because they are based on divergence measure that increases when the summary's quality is low and decreases when its quality is high.

3.6 Conclusion

We introduced wikiSERA, an open-source system for summary evaluation that is domain-independent. wikiSERA is an improved version of SERA, where we redefine query reformulation based on POS Tags distribution of datasets issued from different domains: AQUAINT-2 (news), PubMed (biomedical), and Wikipedia (general domain). wikiSERA outperforms SERA and reduces its gap with ROUGE for TAC 2008, while it outperforms it in some cases on Wikipedia with TAC 2009.

Unsurprisingly, the comparison with evaluation methods that do not rely on human references reveals a large gap in favor of wikiSERA since it relies on human references. In contrast, the other two baselines do not. This finding is intuitive insofar wikiSERA exploits human references while the other two baselines do not.

Moreover, we extensively study the performance of SERA and wikiSERA by computing its correlation with individual scores of human annotators and the average score of each combination of three annotators and all the annotators. Based on the conducted experiments, we note that each annotator presents different correlations with Pyramid and Responsiveness. Consequently, it is sometimes better to use only the human annotators with the highest correlation since using the average score of the four of them affects the obtained results.

In addition, our extensive study includes building many indexes with a different number of documents from AQUAINT-2, Wikipedia, TAC 2008, and TAC 2009 datasets. We build seven indexes with AQUAINT-2 (with sizes of 10000, 15000, 30000, 60000, 89760, 179520, and 825148 documents). We get the best correlation from the indexes with few documents such as 10000, 15000, and 30000 documents. Moreover, according to the indexes built with TAC 2008 and TAC 2009 documents, we note that the correlations are lower than those obtained when built indexes contain at least 10000 documents from AQUAINT-2. Based on the above experience, we lead experiments using 10000, 15000, 30000, and all the documents from Wikipedia documents. We outperform the Pearson correlation compared to ROUGE when indexing 10000 documents. Consequently, we conclude that the ideal number of documents to build the index is 10000 in terms of execution time and performance.

Finally, we conduct experiments on CNNDM to assess the quality of different evaluation systems on both extractive and abstractive summaries. Based on the correlation scores obtained with: (1) both extractive and abstractive systems (Table 3.3-$left$), (2) only extractive summaries (Table 3.3-*middle*), and (3) only abstractive summaries (Table 3.3-*right*), we conclude that wikiSERA is a reliable automatic evaluation measure for evaluating automatic summaries. According to Bhandari et al. (2020), it is important to use a different evaluation metric for each dataset. However, based on experiments we led on TAC 2008, TAC 2009, and CNNDM, we prove that it is possible to have one robust metric to evaluate all types of summaries (extractive and abstractive) from all domains. Interestingly, wikiSERA outperforms not only JS-2 but also the other two recent methods (BERTscore and MoverScore). It is also the only evaluation method that consistently approaches ROUGE's correlations and even exceeds it in one case on TAC 2009.

Chapter 4

Automatic Summarization of Long Medical Texts

4.1 Introduction

Text summarization is the task of generating summaries from a source text. There are two main families of text summarization approaches: extractive and abstractive. On the one hand, extractive methods select the most relevant sentences from the input text and concatenate them to obtain the summary. On the other hand, abstractive approaches aim to generate summaries as humans do by paraphrasing the most crucial sentences and possibly generating novel words.

Intuitively, extractive summarization is easier than abstractive one. For this reason, most of the first research approaches were focused on extractive techniques (Luhn, 1958, Jing and McKeown, 1999). Over the years, abstractive summarization has been gaining momentum with the introduction of RNNs (*Recurrent Neural Networks*). In particular LSTMs (Hochreiter and Schmidhuber, 1997), GRUs (Cho et al., 2014), and most recently, Transformers Neural Networks (Vaswani et al., 2017).

In abstractive summarization, sequence-to-sequence models use encoder-decoder architectures. The first models were based on RNNs, and they obtain summaries at the sentence level (Rush et al., 2015, Nallapati et al., 2016).

In the last years, the interest in working on abstractive summarization has been increasing. In the beginning, sequence-to-sequence models focused on RNN-based architectures (See et al., 2017, Nallapati et al., 2017). However, most recently, lots of research surged using Transformer Neural Networks (Narayan et al., 2018a, Fabbri et al., 2019, Zhang and Tetreault, 2019) with and without pre-training. The latter setting gained attraction recently in the research community (Lewis et al., 2019, Kim et al., 2019, Zhang and Tetreault, 2019, Zhang et al., 2020a) to make use of previously learned powerful representations to improve summary generation.

Consequently, the interest to collect a variety of datasets increased as well. For instance, the most recent corpora include larger texts than old ones, where documents result from various domains compared to old corpora, where the news domain was the predominant type of articles.

In this thesis, we introduce $HazPi$, an improved Transformer architecture for abstractive text summarization in which we propose two contributions. The first one is to use a multi-encoder to process parallelly long input sequences. The second contribution is to add an extra training stage inspired by Hoang et al. (2019), where we propose an end-chunk task training. This extra training phase consists of feeding the reference summary chunk by chunk to the decoder instead of the token-by-token technique proposed by Hoang et al. (2019).

We evaluate our proposed architecture on a medical dataset built by Cohan et al. (2018), and conducted experiments to show that: (1) training is faster when using more than one input encoder, (2) generated summaries are learned efficiently and relatively fast when using the chuck-by-chunk decoding. $HazPi$ achieves higher scores than the traditional Transformer NN with and without pre-training the model.

To summarize our contributions:

- We propose $HazPi$, an improved Transformer (Vaswani et al., 2017) neural network, where we use four encoders instead of one (Subsection 4.2.1).
- Inspired by Hoang et al. (2019), we propose an extra end-chunk task training step in order to improve quality of summaries generated by the system (Subsection 4.2.2).

4.2 Proposed approach: HazPi

4.2.1 The multi-encoder Transformer model

Inspired by the original Transformer (Vaswani et al., 2017) (called $Transformer_{ORIGINAL}$ here), we propose $HazPi$, a modified Transformer architecture consisting of four encoders instead of one. Furthermore, $HazPi$ adds a second phase of training. In the following lines, we present these two contributions.

We consider each article in the training set as a sequence of tokens. Before feeding this sequence of tokens to the model, we distinguish between two cases: (1) if the text sequence contains 2,000 tokens or more, we truncate it to its first 2000 tokens; (2) if the text sequences contains less than 2,000 tokens, we add special padding tokens until the sequence achieves a length of 2,000 tokens. We perform the same process for the gold standard until it achieves a length of 216 tokens, as shown in Figure 4.1.

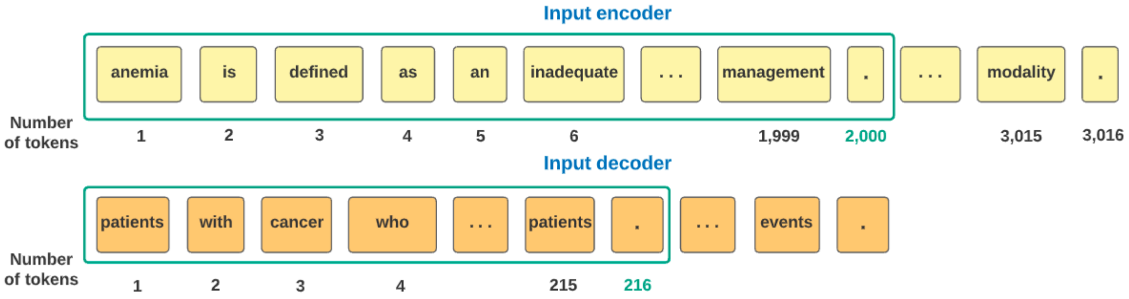


Figure 4.1: Truncating input text and gold abstract in *HazPi*

Once we have the 2,000 tokens per sequence, we split the sequence into four equal chunks of text, where each chunk contains 500 tokens. As already mentioned, *HazPi* consists of four encoders. Therefore, we feed each text chunk into a different encoder (Figure 4.2), where each encoder processes one text chunk. Once the encoders process the chunks, we concatenate the four encoders' outputs before feeding the resulting matrix to the decoder.

Contrarily to $Transformer_{ORIGINAL}$ that uses one encoder with eight self-attention heads for the whole input, we use four encoders with eight self-attention heads each (see Figure 4.3). This choice is motivated by works such as Fabbri et al. (2019) and Zhang et al. (2020a) that achieved competitive results compared to state of the art by reading sequences of length $L_{input} = 500$ and $L_{input} = 512$ tokens as input for the encoder, respectively. Since we experiment with sequences that are 2000-tokens long, it is fair to use four encoders, each having $L_{input} = 500$ tokens.

This improvement tries to cope with Transformer's attention under-performance with long input sequences. Four encoders with eight multi-attention heads each would improve the Transformer processing of long sequences and reduce the training time without penalizing the quality of generated summaries (see Section 4.4). As it is clearly stated in the original Transformer paper (Vaswani et al., 2017), self-attention models tend to perform better with shorter token sequences. Thus, we hypothesize that splitting long sequences and distributing them in the multi-encoder layer would reduce the training time and improve long source documents' processing. Our goal here is to measure the impact of a substantial modification in the original Transformer architecture with and without pre-training and then compare it with the state-of-the-art PEGASUS approach (Zhang et al., 2020a).

Figure 4.3 shows multi-head attention layers in each of the four encoders of our

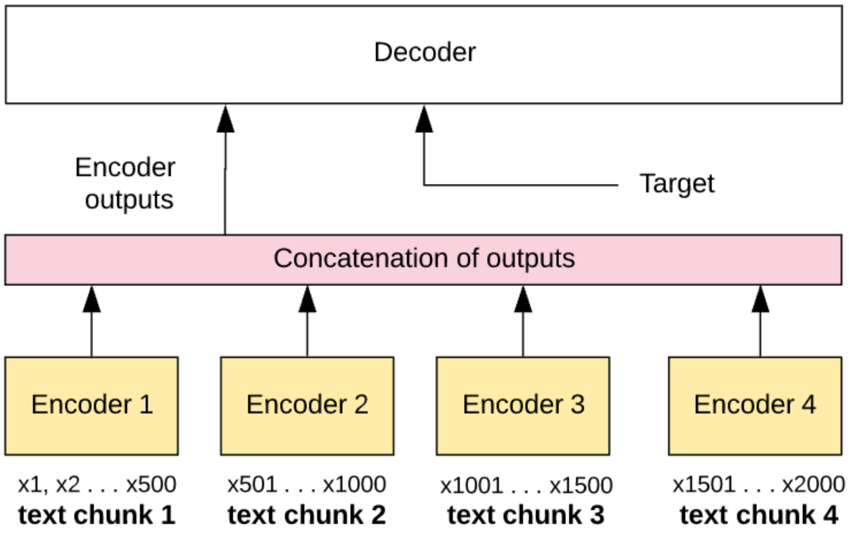


Figure 4.2: Concatenation of the four encoders in *HazPi*

multi-encoder architecture. This input of multi-head attention encoders is a slice of the source text's embeddings and positional encodings. This input can be linearly transformed to get queries (Q), keys (K), and values (V) matrices from the original mono-encoder Transformer model. Each attention head uses different linear transformations to represent words. For this reason, different heads can learn different relationships between words. Consequently, using a multi-encoding schema with four encoders, our model contains 32 attention heads compared to 8 in the original Transformer. We hypothesize that this increase in the number of attention heads would lead our multi-encoder Transformer to learn different relationships between words. While most of the Transformer models read 512 or at most 1024 (Zhang et al., 2020a) tokens, we use input sequences having $L_{input} = 2000$ tokens, which is closer to the average size of 3016 tokens of medical documents from the PubMed dataset (Cohan et al., 2018). However, this multi-encoding schema is only applied at the encoder level, not at the decoder level, where sequences are short enough to generate summaries of the desired length (216-tokens long).

4.2.2 End-Chunk Task Training (ECTT)

End task training was introduced by Hoang et al. (2019). The goal of their approach was to adapt a generic pre-trained text generation Transformer to the ATS task. End task training is an additional training step that aims to constrain the neural network to maximize the log-likelihood probability of generating a pertinent summary given

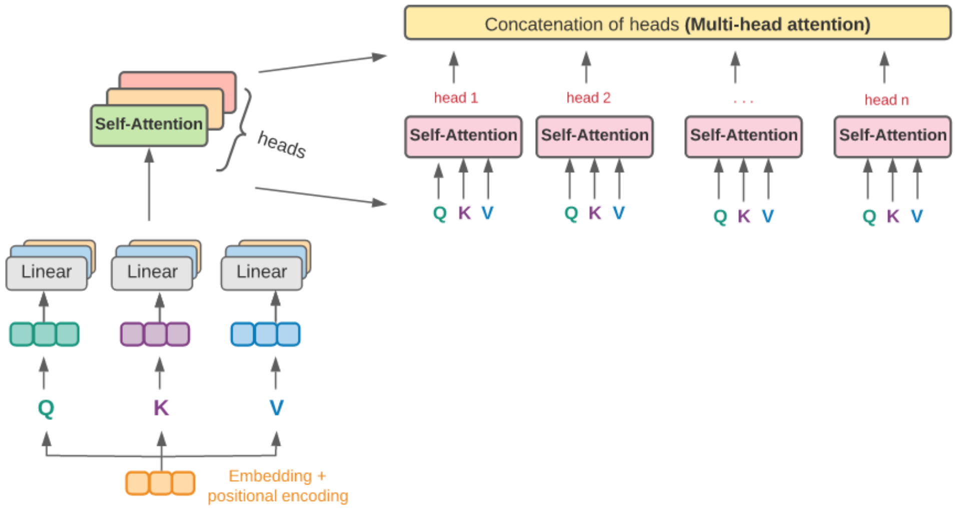


Figure 4.3: Multi-head attention in each encoder

the reference summary. We adapted the equation from Hoang et al. (2019) to receive sequences of tokens (or chunks) instead of a token-by-token flow. We call this improvement *end-chunk task training (ECTT)*. While the approach of Hoang et al. (2019) feeds increasing 1-token differential sentences, we feed chunks of cs (chunk size) tokens. The new loss function $\mathcal{L}_{ECTT}$ is provided in Equation 4.1.

$$\mathcal{L}_{ECTT} = -\sum_{i=0}^{cn-1} log\ P(\{x^s\}_0^{(i+1)\times cs-1} | \{x^a\}_0^{M-1}) \tag{4.1}$$

where:

- M is the number of tokens in the article
- cn is the number of chunks into which the summary is divided
- cs is the chunk size, such that $cs = N/cn$, where N is the number of tokens in the summary
- x^s is a token from the summary
- x^a is a token from the article
- $\{x^s\}_0^{(i+1)\times cs-1}$ is the summary chunk, counting from the first token (0) until the token number $(i+1)\times cs-1$
- $\{x^a\}_0^{M-1}$ is the input article

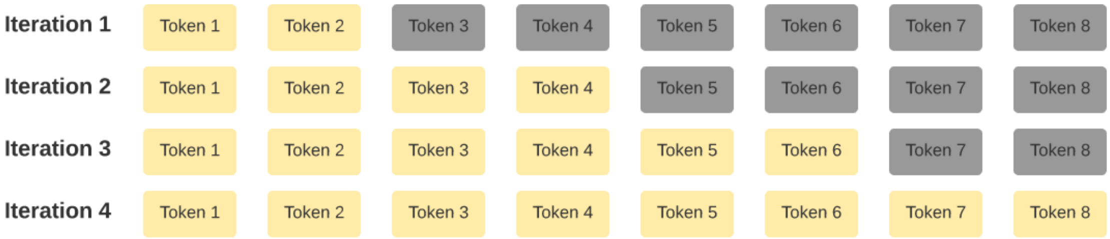


Figure 4.4: Example of how the decoder is fed by chunks of text progressively

Figure 4.4 shows an example of how the decoder is fed by chunks of text progressively. Assuming that we have a sequence of 8 tokens, and each chunk of text consists of 2 tokens, we have four chunks. Therefore, in the first iteration, we feed the decoder with two tokens. In the second iteration, the decoder is fed with four tokens, and the process is repeated until iteration four, where the whole sequence of text is fed to the decoder. We prove the effectiveness of ECTT experimentally in Subsection 4.4.1 below.

4.3 Experimental framework

4.3.1 Datasets

We conduct our experiments using two experimental protocols: with and without pre-training. In a non-pre-training protocol, the model is randomly initialized before being trained on the target dataset. On the contrary, in a pre-training protocol, the model is first pre-trained on a large source dataset before being updated with articles from the target dataset. Pre-training the model on a sufficiently large source dataset from the same domain of the target dataset helps accelerate the training process and improve the model's generalization (Zhang et al., 2020a). We present hereafter the source and target datasets used in our experiments.

- **Target dataset** - we use PubMed, a dataset collected by Cohan et al. (2018) from the well-known PubMed scientific repository (`PubMed.gov`). This dataset comprises 130,397 documents, where 117,108 are in the training set, 6,631 are in the validation set, and 6,658 are in the test set. We use the validation set to tune hyper-parameters during the training process and the test set to get the final summaries.

- **Source dataset** - In the pre-training protocol, we pre-train our models on a dataset (called CovMed below) that we built by mixing articles from the Covid-19 (Coronavirus) dataset built by the White House (House, 2020) and articles from PubMed that are different from those used in the target dataset. In total, this dataset contains 646,960 articles, where 549,902 are in the training set, 32,348 are in the validation set, and 32,362 are in the test set. CovMed is thus five times larger than the target dataset and has, on average, 4364 tokens and 156 sentences in articles, 301 tokens and 11 sentences in the summaries.

4.3.2 Baselines

Experiments were conducted with many strong baselines described as follows:

- **Transformer$_{\mathbf{ORIGINAL}}$** - is a mono-encoder architecture described in Subsection 2.4.6.3 of Chapter 2. Inspired by Gehrmann et al. (2018), we use 4 layers contrarily to the initially proposed 6-layers architecture (Vaswani et al., 2017) for memory occupation issues.

- **PEGASUS** - is a novel Transformer-based approach with a new self-supervised objective (Zhang et al., 2020a). Authors use very large models pre-trained on massive text corpora (see Subsection 2.4.6.3 of Chapter 2). We compare *Hazpi* with three variants of this system:

  - **Transformer$_{\mathbf{BASE}}$** - The architecture of this model has $L = 12$, $H = 768$, $F = 4096$, and $A = 16$, where L is the number of layers in the encoder and the decoder (i.e. Transformer blocks), H is the hidden size, F is the feed-forward layer size, and A is the number of self-attention heads.

  - **PEGASUS$_{\mathbf{LARGE}}$** - is a larger version of $Transformer_{BASE}$, where authors use $L = 16$, $H = 1024$, $F = 4096$, and $A = 16$. Note that this system is pre-trained on the $C4$ (Raffel et al., 2020) corpus that contains 350 million texts of size 750GB extracted from the web.

- **LSTM$_{\mathbf{AWARE}}$** - is an ATS system proposed by Cohan et al. (2018). It is based on a hierarchical encoder to model the discourse structure of documents and a discourse-aware decoder to generate summaries. Both the encoder and the decoder are implemented as Long Short Term Memory (LSTM) networks.

- **GTTP** - (Get To The Point) is an extraction-abstraction hybrid system introduced by See et al. (2017). It is based on a pointer-generator network and a coverage model. The pointer copies factual information, while the generator paraphrases passages from the source text. The coverage model's role is to keep track of what has been generated so far in order to avoid repetitions.

4.3.3 Implementation details and evaluation methodology

Implementation - We implemented $Transformer_{ORIGINAL}$ and $HazPi$ (our multi-encoder Transformer) in Python[1] using Keras (Chollet et al., 2015). We used the same parameters and authors' public implementations to run $LSTM_{AWARE}$ and $GTTP$ from scratch and model checkpoints from HuggingFace[2] to generate summaries from the $PEGASUS$ variants. We trained our models on 8 GPUs of Nvidia Quadro P5000 with 16GB of RAM capacity each.

Hyper parameters - We truncate scientific documents to their first 2000 tokens to evaluate our model's performance on long sequences. Generated summaries are of size $L_{output} = 216$, and only the most frequent 100,000 tokens are kept in the vocabulary. Compared to baselines, we use the same summary length than Cohan et al. (2018). However, See et al. (2017) and Zhang et al. (2020a) generate 100 and 256 tokens-long summaries, respectively.

Following Vaswani et al. (2017), we use Adam algorithm for optimization (Kingma and Ba, 2015) with $batch\ size = 32$, $\beta_1 = 0.9$ and $\beta_2 = 0.98$. We vary the learning rate using the formula provided in Equation 4.2.

$$lr = d_{model}^{-0.5} \cdot min(step\_num^{-0.5}, step\_num \cdot warmup\_steps^{-1.5}) \tag{4.2}$$

Following this formula, the learning rate increases linearly for the first $warmup\_steps = 4000$ training steps and decreases later proportionally to the inverse square root of the step number ($step\_num$).

In the non-pre-training protocol, we train $Transformer_{ORIGINAL}$ and $HazPi$ for 300 epochs. In the pre-training protocol, we train them for 350 epochs on the source dataset and later fine-tune them for 200 epochs on the target dataset. The end-chunk extra training stage lasts 10 epochs with and without pre-training, and the optimal chunk size in this stage was empirically set to $cs = 27$ tokens. Besides,

[1] Our code is available at `https://github.com/JessicaLopezEspejel/HazPi/` and is partially based on that in `https://github.com/rojagtap/abstractive_summarizer/`

[2] `https://github.com/huggingface/transformers/`

we use beam search with a beam size $k = 6$ and $\alpha = 0.8$ for the length-penalty technique described in Subsection 2.4.6.4 of Chapter 2.

Evaluation - To assess the performance of all tested methods, we use three evaluation methods: (1) ROUGE (Lin, 2004) (based on lexical overlaps between tokens and phrases in the generated summary and the gold-standard one), (2) SERA (Cohan and Goharian, 2016) (based on the semantic content analysis between the generated summary and the gold-standard one), and (3) wikiSERA (our proposed evaluation approach presented in Chapter 3). We use all variants of SERA and wikiSERA for evaluation. However, we only report scores with ROUGE-1 (unigrams), ROUGE-2 (bigrams), and ROUGE-L (longest common sub-sequence), the most popular ones in the literature.

We also compare the training time between $HazPi$ and $Transfomer_{ORIGINAL}$ to assess the merits of using a multi-encoder schema. Note that we do not compare the training time with other tested methods since they are run on different hardware.

4.4 Results and discussion

In this section, we present obtained results with and without pre-training. In Subsection 4.4.1, we first discuss results in terms of ROUGE, the lexical-based evaluation method. Here, we establish a comparison between $Transformer_{ORIGINAL}$ and $HazPi$ to study the impact of using multi-encoding. We notably compare their execution time and performance with and without pre-training the model. Second, we compare $HazPi$ with $Transformer_{BASE}$ (the baseline used in the $PEGASUS$ paper (Zhang et al., 2020a) without pre-training) as well as $GTTP$ and $LSTM_{AWARE}$, two non-pre-trained LSTM-based approaches. Finally, we establish a comparison with another competitive state-of-the-art approach (Zhang et al., 2020a) that is pre-trained on huge corpora.

In Subsection 4.4.2, we discuss obtained results in terms of SERA and wikiSERA, the semantic content-based evaluation approaches. Here, we present one large table with all variants of the two methods, and we compare $HazPi$ with works from literature with and without pre-training. Note that all methods used for comparison are abstractive except for $GTTP$, a hybrid extractive-abstractive system.

4.4.1 Evaluation with ROUGE

4.4.1.1 Results without pre-training

The left part of Table 4.1 provides results without pre-training of $HazPi$, $Transformer_{BASE}$ from the PEGASUS paper (Zhang et al., 2020a) and $Transformer_{ORIGINAL}$ from Vaswani et al. (2017). Note that $Transformer_{ORIGINAL}$ is implemented with the End-Chunk Task Training (ECTT) that we propose. The reason behind this choice is to assess the merits of using many input encoders.

Performance of Transformer-based approaches - Results show that $HazPi$ outperforms $Transformer_{ORIGINAL}$ in terms of all ROUGE variants. Indeed, it gets 1.41, 0.57, and 0.42 points more than $Transformer_{ORIGINAL}$, respectively. The increase in ROUGE-1 is larger and is intuitive insofar ROUGE-1 is based on the overlap of uni-grams between the candidate summary and the gold standard. Thus, it is easier to satisfy than bi-grams (ROUGE-2) or the longest common subsequence (ROUGE-L). Interestingly, $HazPi$ outperforms the $Transformer_{BASE}$ in terms of ROUGE-1 and ROUGE-2 scores, while it gives slightly lower scores than it in terms of ROUGE-L. Note that $Transformer_{BASE}$ is much deeper than $HazPi$ since it contains 12 layers with 16 self-attention heads.

	Transformer-based				LSTM-based	
	$Transformer$	$Transformer$	$HazPi$	$Transformer$	$GTTP$	$LSTM$
	$ORIGINAL$	$ORIGINAL$ + ECTT	(ours)	$BASE$		$AWARE$
Num. of encoders	1	1	4	1	×	×
Num. of layers	4	4	4	12	×	×
L_{input}	**2000**	**2000**	**2000**	512	400	**2000**
L_{output}	216	216	216	256	100	216
Training time (epoch)	×	58mn21s	**33mn15s**	×	×	×
ROUGE-1	31.49	32.7	**34.11**	33.94	35.86	**38.93**
ROUGE-2	6.37	7.11	**7.68**	7.43	10.22	**15.37**
ROUGE-L	16.97	18.14	18.56	**19.02**	29.69	**35.21**

Table 4.1: ROUGE scores of $Transformer_{ORIGINAL}$, $HazPi$ (ours), $Transformer_{BASE}$ (from the PEGASUS paper), $GTTP$, and $LSTM_{AWARE}$ without pre-training. Note that L_{input} and L_{output} refer to the length of the input sequence and the generated summary, respectively. Best results are in bold.

For the second stage of training (called End-Chunk Task Training), learning chunk by chunk helps to feed information progressively and relatively fast to the decoder. This training is inspired by how humans get knowledge incrementally over time.

Execution time - Our $HazPi$ model runs almost two times faster than the original Transformer baseline ($Transformer_{ORIGINAL}$). According to Table 4.1, one epoch takes $33mn15s$ in $HazPi$ while it takes $58mn21s$ in $Transformer_{ORIGINAL}$. The reason behind this is that our model can capture the most important tokens from the input. This finding is intuitive insofar we increase the number of head attentions when using more than an encoder at a time. In general, each encoder has eight self-attention heads. However, in our experiments, we have four encoders, leading to a total of 32 self-attention heads. Therefore, with more self-attentions, the encoders capture more information from the large input sequences. As mentioned before, we do not compare the execution time of $HazPi$ and $Transformer_{ORIGINAL}$ with other methods since they are not run on the same hardware.

Performance of LSTM-based approaches - The right part of Table 4.1 shows results obtained with $GTTP$ and $LSTM_{AWARE}$, two non-pre-trained LSTM-based approaches. Interestingly, $LSTM_{AWARE}$ provides the best results in all tested methods while reading 2000 tokens as input. $GTTP$ provides good results yet lower than those of $LSTM_{AWARE}$ with 3.07, 5.13, and 5.52 points of ROUGE-1, ROUGE-2, and ROUGE-L, respectively.

Even though Transformer neural networks outperform LSTMs in many NLP tasks, they still lag behind them in terms of ROUGE scores when it comes to the ATS task. Surprisingly, $LSTM_{AWARE}$ outperforms the best results obtained with Transformer-based approaches with 4.81, 7.69, and 16.19 points of ROUGE-1, ROUGE-2, and ROUGE-L, respectively. More discussion is provided in Subsection 4.4.2 of this chapter.

Input sequence length - Our model can read sequences of length $L_{input} = 2000$ tokens while increasing scores of $Transformer_{ORIGINAL}$. This number is closer to the average size of a standard biomedical article (3016 tokens in Cohan et al. (2018)) than the input length used by other Transformer-based state-of-the-art approaches (Zhang et al., 2020a). For instance, $Transformer_{BASE}$ in Table 4.1 uses sequences of length $L_{input} = 512$. To the best of our knowledge, there is no Transformer-based ATS system able to provide good performance using such a large length in the biomedical domain. The longest input text used in literature is $L_{input} = 1024$ tokens by $PEGASUS_{LARGE}$ (Zhang et al., 2020a), the largest pre-trained model used in that paper.

The LSTM-based approach (Cohan et al., 2018) (called $LSTM_{AWARE}$ in Subsection 4.3.2) is still the only ATS system that processes large sequence of length $L_{input} = 2000$ tokens, while getting high ROUGE scores as discussed above. Unfortunately, the input length ($L_{input} = 400$) used in $GTTP$ is very low compared to the length used by other systems. In fact, it is even comparable with the summary length ($L_{output} = 256$) generated by the $Transformer_{BASE}$.

Effect of the ECTT training phase - The effect of the ECTT loss function in the second training stage is to help the neural network to learn the text content progressively. Since the text is passed to the decoder chunk by a chunk, the neural network can learn little by little the gold standard. Therefore, the same text chunk is seen multiple times by the decoder, where the older the chunk, the more it is seen by the network.

We remind that our generated text sequences are 216 tokens long. Since we divide each token sequence into eight chunks, each text chunk contains 27 tokens. In the first iteration, the decoder is fed with the first 27 tokens. In the second iteration, the decoder is fed with the first $27 \times 2 = 54$ tokens, etc. This process is repeated eight times until the whole gold standard is consumed, and all of this happens within one training epoch. Experimentally, we observe how the loss function decreases as the decoder is progressively fed with each of the text chunks. In terms of ROUGE scores, Table 4.1 shows that using the ECTT on top of the classical training of $Transformer_{ORIGINAL}$ yields a gain of 1.21, 0.74, and 1.17 points in terms of ROUGE-1, ROUGE-2, and ROUGE-L, respectively.

4.4.1.2 Results with pre-training

We pre-trained our model using CovMed, the source dataset described in Section 4.3.1. Afterward, we fine-tuned it on the target dataset described in the same section. Table 4.2 is divided in two parts. The upper part shows results obtained with $Transformer_{ORIGINAL}$ (+ECTT) and $HazPi$ with and without pre-training. The lower part provides results of $PEGASUS_{LARGE}$, the largest Transformer-based pre-trained system used in Zhang et al. (2020a).

Effect of pre-training on HazPi and the baseline - Experiments show that pre-training $HazPi$, and $Transformer_{ORIGINAL}$ on a large medical source dataset is

System	L_{input}	Pre-training corpus	ROUGE-1	ROUGE-2	ROUGE-L
$Transformer_{ORIGINAL}$ (Vaswani et al., 2017) + ECTT	**2000**	×	32.70	7.11	18.14
		CovMed (ours)	33.52	7.21	18.32
$HazPi$ (ours)	**2000**	×	34.11	7.68	18.56
		CovMed (ours)	**36.05**	**8.11**	**18.98**
$PEGASUS_{LARGE}$ (Zhang et al., 2020a)	1024	C4	**45.49**	**19.90**	27.69

Table 4.2: ROUGE scores of $Transformer_{ORIGINAL}$, $HazPi$ (ours), and $PEGASUS_{LARGE}$ with pre-training. Best results are in bold.

beneficial in both cases. For the baseline ($Transformer_{ORIGINAL}$), the ROUGE-1, ROUGE-2, and ROUGE-L scores increase by 0.82, 0.1, and 0.18 points, respectively. The increase of scores for $HazPi$ corresponds to 1.94, 0.73, and 0.42 points, respectively. Unsurprisingly, pre-training a model on a large dataset belonging to the same domain helps to start the training process and augment the model's representation capacity.

Comparison with Transformer-based approaches Since $GTTP$ and $LSTM_{AWARE}$ are based on different neural network architecture than $HazPi$, we focus our attention on the detailed comparison of our model with T5 (Raffel et al., 2020) and $PEGASUS_{LARGE}$ (Zhang et al., 2020a). Similarly to us, the latter systems are based on a Transformer architecture with an encoder and a decoder, unlike language models such as BERT (Devlin et al., 2019) that has only an encoder. The difference between $HazPi$ and these two approaches comes mainly from:

- *Size of the source dataset* - Unlike T5 and $PEGASUS_{LARGE}$ that use large pre-training corpora from the news domain, we use a relatively small dataset specialized in medical articles. Our CovMed source dataset contains 646,960 articles (occupying 17.2GB). However, both T5 and PEGASUS use the Colossal Clean Crawled Corpus (C4) dataset, which comprises 350 million texts extracted from the web (occupying 750GB). This huge gap in size between our CovMed source dataset and the one used in $PEGASUS_{LARGE}$ largely explains the gap of results between this approach and $HazPi$. The memory footprint occupied by our source dataset is 2.3% of that occupied by C4. Consequently, the negligible memory requirements that we need make $HazPi$ more interesting when working in a limited storage framework.

- *Domain of the source dataset* - According to Keskar et al. (2019), Huang et al. (2019), and Radford et al. (2019), pre-training a model on a very large dataset containing articles from various domains increase the model's representation and its capacity to summarize heterogeneous texts. In our case, we are not interested in developing a universal system, but one specialized in the medical domain, and the most convenient is to pre-train it on a larger medical dataset than the one used in our experiments.

- *Complexity of the neural network* - The number of layers in a neural network is another relevant criterion that defines the complexity of the model and thus its computational impact. T5 consists of 12 layers both in the encoder and the decoder. Meanwhile, $PEGASUS_{BASE}$ and $PEGASUS_{LARGE}$ (Zhang et al., 2020a) contain 12 and 16 layers, respectively. This is not the case of $Transformer_{ORIGINAL}$ and $HazPi$, where we use four layers. According to Liu et al. (2019), unlike LSTMs, Transformer neural networks capture the semantic of words in intermediate layers. This explains the high results obtained with $PEGASUS_{LARGE}$ in Table 4.2, but more importantly, it explains why $HazPi$ gets ROUGE-1 scores that are closer to those from the state of the art, while its performance lags for the overlap of bigrams (ROUGE-2) and long common sub-sequences (ROUGE-L).

4.4.2 Evaluation with SERA and wikiSERA

Even though ROUGE is the most popular metric used in literature to evaluate summaries automatically, it is based on lexical overlaps and is thus not fair to evaluate abstractive summaries. As already explained, an abstractive summary is made by paraphrasing the source document with possibly novel words such as synonyms.

In Table 4.3, we present results obtained with and without pre-training of approaches discussed above in terms of SERA and wikiSERA. These two approaches are more convenient to evaluate abstractive summaries because they are based on the semantic analysis between the generated summary and its goal standard. Note that, contrarily to Chapter 3, we used as an index of wikiSERA ten thousand medical documents selected randomly from our CovMed dataset instead of Wikipedia. For the sake of comparability, we use the same index to run the SERA method.

	without pre-training			with pre-training		
	$GTTP$	$Transformer$ $ORIGINAL$ + ECTT	$HazPi$ (ours)	$Transformer$ $ORIGINAL$ + ECTT	$HazPi$ (ours)	$PEGASUS$ $LARGE$
SERA-5	20.38	26.96	**27.99**	27.29	28.35	**38.47**
SERA-10	21.67	28.94	**29.71**	29.29	30.61	**39.90**
SERA-NP-5	20.36	25.25	**26.15**	25.67	26.42	**37.15**
SERA-NP-10	21.74	27.78	**28.69**	28.16	29.42	**39.24**
SERA-KW-5	20.48	26.98	**27.98**	27.40	28.51	**38.79**
SERA-KW-10	21.75	28.91	**29.63**	29.35	30.74	**40.21**
SERA-DIS-5	14.59	18.14	**22.61**	18.54	23.12	**26.95**
SERA-DIS-10	12.96	16.06	**16.61**	16.32	16.95	**23.25**
SERA-DIS-NP-5	14.54	16.97	**17.61**	17.17	18.85	**25.87**
SERA-DIS-NP-10	12.97	15.28	**15.88**	15.54	16.48	**22.65**
SERA-DIS-KW-5	14.65	**19.52**	18.14	19.03	18.57	**27.17**
SERA-DIS-KW-10	13.03	**22.61**	12.61	16.34	17.47	**23.44**
wikiSERA-5	20.51	26.17	**27.50**	26.86	28.84	**38.08**
wikiSERA-10	21.95	28.41	**29.39**	29.03	30.24	**39.83**
wikiSERA-DIS-5	14.77	17.56	**18.53**	18.01	19.34	**26.57**
wikiSERA-DIS-10	13.14	15.66	**16.34**	16.03	17.15	**23.19**

Table 4.3: SERA and wikiSERA scores of $GTTP$, $Transformer_{ORIGINAL}$, $HazPi$, and $PEGASUS_{LARGE}$. Best results are in bold.

4.4.2.1 Results without pre-training

Results show that GTTP (See et al., 2017), the LSTM-based ATS system obtains largely lower scores compared to $Transformer_{ORIGINAL}$ (Vaswani et al., 2017) and $HazPi$ without pre-training. On the one hand, the gap in scores between $GTTP$ and $Transformer_{ORIGINAL}$ (Vaswani et al., 2017) varies between 2.31 and 9.58 points obtained with SERA-DIS-NP-10 and SERA-DIS-KW-10, respectively. On the other hand, the gap in scores between $GTTP$ and $HazPi$ varies between 2.91 and 8.04 points obtained with SERA-DIS-NP-10 and SERA-10, respectively. Note that $HazPi$ achieves slightly lower results than $GTTP$ for SERA-DIS-KW-10, where the gap between both is equal to 0.42 points.

As mentioned before, we use the same authors' parameters to generate summaries with See et al. (2017). Therefore, summaries generated by this system contain 100 tokens contrarily to $HazPi$ that generates summaries that are 216 tokens long. This difference in summary length can explain in part the gap of scores between the two approaches. However, according to the state of the art (Zhang et al., 2020a), Transformer neural networks are capable of producing abstractive summaries more effectively than LSTMs. Similar to the scores obtained with ROUGE, $HazPi$ improves the results of $Transformer_{ORIGINAL}$ in most cases, where the gap between both varies between 0.6 and 4.46 points obtained with SERA-DIS-NP-10 and SERA-DIS-5, respectively. Surprisingly, $Transformer_{ORIGINAL}$ outperforms $HazPi$ with 1.37 and 10 points in terms of SERA-DIS-KW-5 and SERA-DIS-KW-10, respectively. This result can be explained by the fact that $HazPi$ generates a low number of keywords than $Transformer_{ORIGINAL}$.

4.4.2.2 Results with pre-training

As already proved with ROUGE results, pre-training on the CovMed dataset is beneficial in most cases for both $HazPi$ and $Transformer_{ORIGINAL}$. On the one hand, the gap between non-pre-trained $Transformer_{ORIGINAL}$ and pre-trained $Transformer_{ORIGINAL}$ varies between 0.2 and 0.68 obtained with SERA-DIS-NP-5 and wikiSERA-5, respectively. On the other hand, the gap between non-pre-trained $HazPi$ and pre-trained $HazPi$ varies between 0.27 and 4.85 obtained with SERA-NP-5 and SERA-DIS-KW-10, respectively.

$PEGASUS_{LARGE}$ achieves the best results compared to $Transformer_{ORIGINAL}$ and $HazPi$ in terms of both SERA and wikiSERA evaluation methods. On the one hand, the gap between $PEGASUS_{LARGE}$ and $Transformer_{ORIGINAL}$ varies between 6.93 and 11.47 obtained with SERA-DIS-10 and SERA-NP-5, respectively. On the other hand, the gap between $PEGASUS_{LARGE}$ and $HazPi$ varies between 3.82 and 10.72 obtained with SERA-DIS-5 and SERA-NP-5, respectively. This finding is intuitive insofar $PEGASUS_{LARGE}$ is four times deeper than the other two approaches (it has 16 layers compared to 4 layers in $HazPi$ and $Transformer_{ORIGINAL}$). Besides, it is pre-trained on C4 that is 43.6 times larger than our CovMed dataset (750GB vs. 17.2 GB). The variation of results obtained with SERA and wikiSERA provides an idea of the POS tag distribution summaries generated by our system. More precisely, summaries generated by $HazPi$ mainly contain words, keywords, and nouns. According to Cohan and Goharian (2016) that based SERA on summaries from the TAC 2014 dataset, the highest correlation in the medical domain is obtained with SERA-NP-5. This finding is confirmed by results obtained with $HazPi$ where our generated medical summaries are better evaluated with SERA-NP-5.

4.4.3 Example of summaries generated by HazPi

Table 4.4 and 4.5 present an example summaries generated by $HazPi$.

Human reference summary
this paper aims to highlight the importance of exercise in patients with rheumatoid arthritis (ra) and to demonstrate the multitude of beneficial effects that properly designed exercise training has in this population . ra is a chronic , systemic , autoimmune disease characterised by decrements to joint health including joint pain and inflammation , fatigue , increased incidence and progression of cardiovascular disease , and accelerated loss of muscle mass , that is , rheumatoid cachexia . these factors contribute to functional limitation , disability , comorbidities , and reduced quality of life . exercise training for ra patients has been shown to be efficacious in reversing cachexia and substantially improving function without exacerbating disease activity and is likely to reduce cardiovascular risk . thus , all ra patients should be encouraged to include aerobic and resistance exercise training as part of routine care . understanding the perceptions of ra patients and health professionals to exercise is key to patients initiating and adhering to effective exercise training .
Summary generated by $HazPi$ (pre-trained)
rheumatoid arthritis (ra) is a chronic autoimmune disease that is associated with an increased risk of cardiovascular disease (cvd) . the purpose of this paper is to review the literature on the role of physical activity in patients with ra . we searched the pubmed , embase , and the cochrane library for randomised controlled trials that assessed the effectiveness of the intervention on cvd and cvd in ra patients . in addition , we reviewed the evidence for their effectiveness in improving the quality of life (qol) of ra in terms of qol .

Table 4.4: Example 1 of a reference summary and its corresponding automatically generated summary with pre-trained $HazPi$

Human reference summary
regmatogenous retinal detachments need prompt intervention particularly when macula is on . unfortunately this is not always easy to ascertain clinically and the chronicity of the event is often muddled in patient ’s histories . developments in optical coherence tomography (oct) have allowed high - resolution axial scans which have enabled the characterisation of retinal changes in retinal detachments . in this paper , we show the changes in retinal morphology observed by spectral domain oct and how this can be used to plan appropriate surgical intervention .
Summary generated by *HazPi* (pre-trained)
the purpose of this paper is to describe the histological findings of retinal detachments in a patient who presented with subretinal haemorrhage . a - old female presented at our clinic with visual acuity of . the patient was treated with pars plana vitrectomy and retinal reattachment was done . spectral - domain optical coherence tomography (sd - oct) was used to assess the changes in retinal morphology and morphology . oct showed a marked reduction in the subretinal and subretinal spaces . this is the first case report of macular detachment in which spectral domain oct has been used for the diagnosis .

Table 4.5: Example 2 of a reference summary and its corresponding automatically generated summary with pre-trained *HazPi*

4.5 Conclusion

In this chapter, we propose two improvements of the Automatic Text Summarization using Transformers (Vaswani et al., 2017). Our approach, called $HazPi$, makes it possible to read long input sequences while reducing training time without penalizing the quality of generated summaries. $HazPi$ is based on a multi-encoder where we split a long input sequence into four chunks and feed each one of them into a different encoder. Our method is also based on an extra training stage where we modify the End Task Training technique proposed by Hoang et al. (2019) to process chunks of tokens when decoding instead of a token-by-token flow.

In our experiments, we use four layers in both the encoder and the decoder. However, most of the Transformer-based architectures in ATS use at least 12 layers. For instance, T5 (Raffel et al., 2020) and $PEGASUS_{BASE}$ use 12 layers, while $PEGASUS_{LARGE}$ (Zhang et al., 2020a) uses 16 layers. Therefore, it would be interesting to increment the number of layers in our model in order to increase its representation capacity.

We evaluate different approaches with ROUGE, the lexical-based approach. We also present results obtained with SERA and wikiSERA, two metrics based on the content relevance of generated summaries. According to obtained results, we confirm using the two automatic measures SERA and wikiSERA, that ROUGE is not fair to evaluate abstractive summaries.

Moreover, conducted experiments measure the performance of different approaches with and without pre-training. Intuitively, the highest results are obtained with pre-trained models where the larger the source dataset, the best is the performance. Indeed, pre-training a model on a large dataset helps to accelerate the training process and improving summary generation.

The neural network depth is another factor that can highly affect a system's performance. $PEGAUS_{LARGE}$ produces abstractive summaries of better quality because it is much deeper than $HazPi$. In fact, a transformer neural network captures the semantic of words at the intermediate layers level. This finding was already confirmed by works such as Liu et al. (2019).

Chapter 5

Conclusions and future work

This thesis is articulated around the Automatic Text Summarization problem, where we aim specifically to summarize abstractively long medical articles. Along with Text Summarization, we tackle the Automatic Summary Evaluation in order to assess the quality of our proposed summarization system. Here, we adapt an evaluation method specialized in biomedical summaries to evaluate summaries from the general domain. The latest adaptation is helpful to evaluate summaries that belong or not to the biomedical domain.

This chapter presents the main conclusions we made from conducted experiments and results obtained in both domains. Furthermore, we discuss potential future directions to open new windows and allow researchers to further improve this work.

5.1 Conclusions and Discussions

5.1.1 Automatic Text Summarization

In Chapter 2, we explained the evolution of several techniques used by the community over time, starting from extractive to abstractive summarization. At the emergence of automatic summarization, the scientific community focused its effort on extractive summarization. Researchers used frequency-based approaches to identify the most relevant tokens and phrases in the source text. Later, attention was put on probabilistic methods that improved the quality of generated summaries until machine learning techniques arose.

Nowadays, deep learning techniques shifted the summarization from the extractive to the abstractive approach. Deep Neural Networks are at present the most powerful models used by researchers. In this thesis, we work on abstractive text summarization that necessitates deep linguistic knowledge to maintain proper language constructs when reformulating text parts. Abstractive ATS is more convenient to handle medical texts that comprise complex and delicate information where simple extraction is not enough.

Proposed method To tackle the ATS task, we decided to adopt the Transformer (Vaswani et al., 2017) neural networks for their ability to process input sequences in parallel. In Chapter 4 of this manuscript, we propose *HazPi*, a modified Transformer architecture that consists of using four input encoders instead of one. Our method aims to reduce training times and allow reading larger input sequences while improving the quality of generated summaries. We used two experimental protocols as follows:

- Without pre-training - Here, we randomly initialize *HazPi* and train it on PubMed (Cohan et al., 2018) target dataset.
- With pre-training - Here, we pre-train *HazPi* on an additional and independent medical dataset before being fine-tuned on our target dataset. We built CovMed, an additional dataset that comprises a mix of articles coming from PubMed (for Biotechnology Information, 2018) repository (different from those used in the target dataset) and articles from the Covid-19 (House, 2020) dataset. In total, this dataset consists of 646,960 pairs of abstracts and articles.

Discussion of obtained results - The comparison with one of the best approaches from the state of the art reveals that our approach is promising, especially in the pre-training protocol. The neural network takes advantage of previously learned information to start the target dataset's training process.

Summaries generated by our system are coherent and readable. In terms of ROUGE scores, we could not outperform $LSTM_{AWARE}$, the LSTM-based summarization system developed by Cohan et al. (2018), while we could largely outperform it using SERA and wikiSERA evaluation approaches that are better suited for abstractive summary evaluation. Note that other systems inspired by Transformer neural networks such as PEGASUS (Zhang et al., 2020a), and T5 (Raffel et al., 2020) could outperform this approach in terms of ROUGE scores.

We present below the main differences between *HazPi* and these two systems in order to investigate the reason behind their performance. Note that we base our discussion about T5 and PEGASUS because, similarly to *HazPi* and unlike architectures such as BERT (Devlin et al., 2019) (which consists only of a single stack of layers), T5 and PEGASUS follow the architecture proposed by Vaswani et al. (2017) (based on an encoder and a decoder).

- *Number of layers* - On the one hand, T5 follows the same size and configuration as BERT (Devlin et al., 2019): it consists of 12 layers both in the encoder and in the decoder. On the other hand, $PEGASUS_{BASE}$ and $PEGASUS_{LARGE}$ contain 12 and 16 layers, respectively. Unfortunately, this is not the case with our architecture based on four layers for memory explosion issues. According to Liu et al. (2019), intermediate layers in a transformer neural network produce more powerful representations for semantic tasks. Therefore, it makes sense that incrementing the number of layers in our architecture would achieve higher ROUGE scores.

- *Size of pre-training dataset* - According to Keskar et al. (2019), Huang et al. (2019), and Radford et al. (2019), pre-training a model on a huge dataset (with at least hundreds of millions of articles) before fine-tuning it on the target dataset helps to improve the training process and generate better summaries. For instance, Raffel et al. (2020) and Zhang et al. (2020a) pre-trained their T5 and $PEGASUS$ models on the Colossal Clean Crawled Corpus (C4) that consists of English texts extracted from the web (350 million web-pages $\equiv$ 750 GB). Unlike these works, our system was pre-trained with a relatively small dataset (646,960 pairs of articles and abstracts $\equiv$ 17.2 GB) for the same reason mentioned above. That is to say, our CovMed source dataset occupies a negligible size of 2.3% of the total size occupied by C4 used to pre-train $PEGASUS$ and T5. To the best of our knowledge, there is no such a huge available medical dataset for summarization. Therefore, building a larger medical dataset is necessary than the one we built in our approach.

5.1.2 Automatic Summary Evaluation

As mentioned before, Automatic Text Summarization is not enough alone. It is crucial to have an automatic evaluation method to assess the quality of generated summaries. We described in Chapter 2 the main approaches used in the automatic evaluation, where we differentiated between extrinsic and intrinsic approaches, as well as manual and automatic approaches. Automatic summary evaluation is essential insofar as it helps improving automatic summarization systems. Humans made the first approach of manual summary evaluation. However, with the continuing increase in the information volume and the emergence of big data, human evaluation became time and money expensive. The need to have automatic evaluation systems independent of any human evaluation is prominent. However, the performance of existing automatic systems is still far from having a high correlation with human evaluation.

Proposed method - In Chapter 3 of this thesis, we highlighted that it is crucial to have an automatic method that is efficient to evaluate abstractive summaries and is not restricted to one domain of application. ROUGE (Lin, 2004) is until now the most popular evaluation method, but it is unfair to evaluate abstractive summaries. SERA (Cohan et al., 2018) was later proposed to tackle this problem. It relies partially on human intervention and is based on a content relevance analysis that considers candidate summaries as queries and searches them in an index built from a large and related dataset of source texts. Unfortunately, SERA was designed to assess the quality of summaries from the biomedical domain only. We started from this motivation and proposed wikiSERA, an improvement of SERA (Cohan et al., 2018) that is domain-independent. For this reason, and based on a POS tag analysis of several corpora belonging to different domains, we redefined query reformulation in SERA to make it generic and efficient to evaluate summaries from all domains. Contrarily to the SERA system that was not publicly shared, wikiSERA is an open-source system ready to be used by the community to evaluate summaries thanks to the index that we built from the Wikipedia public encyclopedia. wikiSERA improved correlation scores of SERA with human references, especially for small-size indexes. In few cases, wikiSERA even outperformed ROUGE (Lin, 2004), one of the most popular evaluation approaches based on lexical overlaps.

Compared to automatic summary evaluation approaches that do not need any human intervention, wikiSERA provides better results in all tested configurations. This finding is intuitive insofar as wikiSERA needs reference summaries for the information retrieval, while the other approaches do not.

Extensive study on query datasets - We led extensive experiments on SERA and wikiSERA, and corresponding results are provided in Appendix A. We notably studied the impact of human annotators on the score correlation of both SERA and wikiSERA. We computed the correlation by taking the score of each human reference individually, and we also averaged scores from two, three, and four (all) human annotators. Results obtained on the TAC 2008 dataset is hard to retrieve a pattern, as the best human annotator changes from one case to another. For the TAC 2009 dataset, the first human annotator achieves the best correlations with Pyramid. In contrast, the third human annotator gets the lowest correlations.

The reasons behind studying extensively TAC 2008 and TAC 2009 datasets are:

- First, both of TAC 2008 and TAC 2009 datasets are news corpora. According to Kryscinski et al. (2020), news-related summarization datasets such as CNN/Daily Mail contain strong layout biases. Therefore, these datasets' evaluation provides a fair idea about each human annotator's performance and the tag distribution in news text.
- Second, a study conducted by Kryscinski et al. (2020) revealed that associating each news article with only a single reference summary leaves the task of summarization under-constrained. This is not the case for TAC datasets since each article is associated with four reference summaries.
- Finally, our election to work with TAC 2008 and TAC 2009 is supported by Gillick and Liu (2010), who showed that summary judgments obtained by experts achieve better performance than using manual annotators from non-experts. In our study, human reference summaries are written by expert journalists.

In addition to the experiments with TAC datasets, we used CNNDM (Bhandari et al., 2020). Contrarily to TAC datasets that contain summaries from extractive systems only, this dataset contains candidate summaries obtained from both extractive and abstractive systems. Thus, CNNDM is helpful to assess the robustness of wikiSERA to evaluate extractive and abstractive approaches.

5.2 Directions for future research

This section presents some future research directions that might help improve our system's current performance both in automatic text summarization and automatic evaluation of summaries.

5.2.1 Automatic Text Summarization

Based on experiments that we led in Chapter 4, we propose the following perspectives:

1. Based on experiments led with and without pre-training and the comparison with the PEGASUS pre-training protocol, the size of the corpus used to pre-train the language model has a huge influence on the quality of summaries

generated by the system. Thus, building a very large medical corpus and using it to pre-train our system could vastly improve its performance.

2. Our system is based on four layers in both the encoder and the decoder. However, according to state of the art, most of the systems use at least twelve layers. Therefore, one direction could be to increment the number of layers in our system. The idea is to keep the number of self-attention heads, and the size of hidden layers fixed while increasing their capacity to learn representations, which might also improve the quality of generated summaries.

3. When it comes to the medical and biomedical articles, we found that the articles are divided into sections. Therefore, there exist two different approaches to select relevant sections as input for our model. The first one is selecting the four most similar sections to the gold summary, where each encoder from the multi-encoder architecture handles a section. In contrast to the current approach where we truncate the whole document to the first 2000 tokens, we could truncate each section to five hundred tokens to read in total two thousand tokens. This way to read the input document will diversify the information read by our model and give more chance to later sections such as the conclusions. The second way to choose input sections is inspired by Liu et al. (2018). It consists of first getting a summary of a medical article using a well-known automatic summarization system that provides long summaries of good quality. The output summary of such a method will be the input of our model. This approach could be seen as an overlaying of two automatic summarization methods, where the second one compresses the summary generated by the first one.

4. In the second stage of the training that we call end-chunk task training, we feed the decoder with the gold summary chunk by chunk (each containing 27 tokens) until we consume all the sequence. An alternative way to proceed is to vary the number of tokens per chunk. More specifically, attempt to read the gold-standard token by token as Hoang et al. (2019) did. The difference with the end-task training done by the latter authors is that we do not adapt a language model designed for another NLP task to text summarization. Instead, we adapt their approach to our task because we hypothesize that our model is able to produce summaries of better quality if we feed the gold summary tokens progressively.

5. Our multi-encoder architecture was trained in an end-to-end manner with the end-chunk task training. However, it would be interesting to build our end-chunk task training on top of other pre-trained summarization systems, such as the PEGASUS' checkpoints, to assess the merits and limitations of our chunk-by-chunk decoding scheme.

5.2.2 Automatic Summary Evaluation

In this section, we present some future directions that can improve the performance of automatic evaluation approaches.

1. We evaluate TAC 2008 and TAC 2009 datasets because they contain texts that belong to the general domain and because each summary has four manual references. However, most of the candidate summaries associated with articles from these datasets were generated from extractive systems. That is why an interesting experiment would be to evaluate SERA and wikiSERA on an extensive corpus where the summaries are generated by abstractive systems, and each summary has more than one manual reference summary.

2. To have human annotators is time and money expensive. This is the main reason why researchers are trying to avoid human references and automatically evaluating candidate summaries. Therefore, an interesting direction is to get the SERA and wikiSERA scores using as queries the article text as the (one and unique) reference summary and its abstract as the candidate summary. Once we have the scores from each summary, we can compute the correlation with human assessments.

3. We analyze the news corpora and the medical dataset to detect which tags are most frequent in each type of text belonging to different domains. Based on the different distribution of tags in each domain, we propose a novel redefine query method called wikiSERA. However, we studied the corpus according to the domain to which each dataset belongs (medical or general domain). However, it can be possible to use different redefinition queries depending on each corpus's tag distribution.

4. We refine the reference summaries and candidate summaries using a specific tag such as nouns and keywords. In other words, we extract from the summaries only the words having to the tag that we are searching for. However, we did not

take into account the context of this term. We believe that if we consider the context of the word extracted from summaries, the evaluation will be better. We can specify a size window to determine how many tokens will be part of the context forward and backward.

Appendix A

Extensive study of different evaluation approaches

A.1 ROUGE

A.1.1 Correlation of ROUGE with Pyramid and Responsiveness on TAC 2008

Method	Pyramid 3M			Responsiveness		
	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
ROUGE-1-F	0.908	0.941	0.787	0.853	0.883	0.702
ROUGE-1-P	0.730	0.841	0.643	0.698	0.803	0.626
ROUGE-1-R	0.911	0.935	0.774	0.851	0.858	0.665
ROUGE-2-F	0.940	0.965	0.843	0.892	0.915	0.746
ROUGE-2-P	0.911	0.942	0.788	0.873	0.901	0.730
ROUGE-2-R	**0.946**	**0.967**	**0.851**	0.894	0.918	0.755
ROUGE-3-F	0.941	0.951	0.810	**0.915**	**0.924**	**0.767**
ROUGE-3-P	0.926	0.934	0.783	0.909	0.918	0.766
ROUGE-3-R	0.945	0.951	0.811	0.914	0.922	0.763
ROUGE-L-F	0.878	0.925	0.756	0.823	0.868	0.689
ROUGE-L-P	0.711	0.823	0.632	0.679	0.794	0.611
ROUGE-L-R	0.882	0.927	0.762	0.823	0.856	0.661
ROUGE-W-1.2-F	0.901	0.940	0.782	0.848	0.878	0.701
ROUGE-W-1.2-P	0.712	0.822	0.631	0.688	0.794	0.620
ROUGE-W-1.2-R	0.897	0.940	0.785	0.841	0.871	0.684
ROUGE-SU4-F	0.917	0.949	0.805	0.870	0.904	0.728
ROUGE-SU4-P	0.839	0.910	0.728	0.805	0.869	0.689
ROUGE-SU4-R	0.927	0.950	0.800	0.874	0.908	0.736

Table A.1: Correlation coefficients, in terms of Pearson, Spearman and Kendall, of ROUGE with Pyramid (using 3 references) and Responsiveness, on TAC 2008.

A.1.2 Correlation of ROUGE with Pyramid and Responsiveness on TAC 2009

Method	Pyramid 3M			Responsiveness		
	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
ROUGE-1-F	0.951	0.915	0.788	0.835	0.793	0.622
ROUGE-1-P	0.923	0.845	0.678	0.791	0.789	0.630
ROUGE-1-R	0.926	0.892	0.748	0.814	0.764	0.591
ROUGE-2-F	0.930	0.955	0.839	0.740	0.831	0.664
ROUGE-2-P	0.906	0.937	0.796	0.716	0.829	0.658
ROUGE-2-R	0.937	0.952	0.841	0.746	0.820	0.654
ROUGE-3-F	0.842	0.964	0.841	0.622	0.852	0.675
ROUGE-3-P	0.828	0.940	0.800	0.610	0.839	0.656
ROUGE-3-R	0.848	0.964	0.845	0.627	0.845	0.673
ROUGE-L-F	0.865	0.604	0.461	0.649	0.414	0.294
ROUGE-L-P	0.801	0.546	0.406	0.573	0.360	0.255
ROUGE-L-R	0.875	0.622	0.474	0.663	0.414	0.298
ROUGE-W-1.2-F	0.882	0.654	0.512	0.651	0.462	0.341
ROUGE-W-1.2-P	0.798	0.514	0.393	0.558	0.337	0.237
ROUGE-W-1.2-R	0.889	0.671	0.529	0.659	0.469	0.340
ROUGE-SU4-F	0.934	0.940	0.818	0.747	0.808	0.639
ROUGE-SU4-P	0.893	0.910	0.761	0.702	0.804	0.638
ROUGE-SU4-R	0.942	0.924	0.787	0.756	0.789	0.619

Table A.2: Correlation coefficients, in terms of Pearson, Spearman and Kendall, of ROUGE with Pyramid (using 3 references) and Responsiveness, on TAC 2009

A.2 SERA and wikiSERA

A.2.1 Correlation of SERA and wikiSERA with Pyramid on TAC2008/AQUAINT-2

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.62	0.676	0.819	0.832	0.863	0.869	0.889	0.844	0.49	0.605	0.789	0.797	0.839	0.863	0.864	0.837	0.343	0.434	0.598	0.614	0.651	0.674	0.684	0.65
	SERA-10	**0.785**	0.803	0.871	**0.891**	0.887	**0.889**	0.886	0.874	**0.718**	0.776	0.859	**0.888**	**0.881**	0.871	0.88	0.876	**0.538**	0.586	0.681	**0.715**	**0.71**	0.695	0.711	0.696
	SERA-NP-5	0.659	0.745	0.751	0.856	0.889	0.848	0.813	0.808	0.585	0.669	0.737	0.831	0.876	0.837	0.79	0.775	0.409	0.484	0.54	0.651	0.704	0.665	0.604	0.595
	SERA-NP-10	0.765	**0.824**	**0.885**	0.868	0.892	0.886	0.893	0.853	0.688	**0.789**	**0.895**	0.835	0.85	**0.875**	**0.899**	0.847	0.503	0.592	**0.724**	0.657	0.665	**0.715**	**0.73**	0.662
	SERA-KW-5	0.625	0.68	0.838	0.844	0.861	0.86	0.889	0.858	0.503	0.609	0.823	0.798	0.841	0.857	0.865	0.816	0.352	0.435	0.628	0.61	0.658	0.67	0.689	0.641
	SERA-KW-10	0.775	0.809	0.873	0.885	0.887	0.888	0.889	0.872	0.691	0.781	0.855	0.876	0.872	0.872	0.884	0.883	0.514	**0.597**	0.677	0.704	0.695	0.688	0.717	0.714
	SERA-DIS-5	0.638	0.673	0.764	0.799	0.847	0.851	0.87	0.859	0.494	0.611	0.743	0.738	0.837	0.837	0.833	0.837	0.345	0.439	0.546	0.551	0.645	0.649	0.658	0.648
	SERA-DIS-10	0.753	0.794	0.844	0.856	0.877	0.873	0.886	**0.889**	0.641	0.764	0.834	0.833	**0.881**	0.836	0.866	**0.89**	0.456	0.567	0.652	0.648	0.695	0.648	0.693	**0.722**
	SERA-DIS-NP-5	0.64	0.736	0.738	0.835	0.857	0.846	0.818	0.78	0.575	0.675	0.741	0.822	0.835	0.843	0.798	0.741	0.39	0.488	0.535	0.624	0.633	0.655	0.594	0.56
	SERA-DIS-NP-10	0.715	0.816	0.849	0.862	**0.895**	0.882	**0.896**	0.831	0.645	0.777	0.863	0.849	0.875	**0.875**	0.894	0.83	0.464	0.567	0.675	0.66	0.695	0.712	0.708	0.647
	SERA-DIS-KW-5	0.641	0.677	0.78	0.812	0.839	0.84	0.867	0.866	0.517	0.611	0.752	0.768	0.811	0.836	0.834	0.827	0.367	0.441	0.552	0.583	0.622	0.647	0.658	0.648
	SERA-DIS-KW-10	0.746	0.802	0.852	0.856	0.876	0.872	0.889	0.885	0.624	0.782	0.838	0.818	**0.881**	0.843	0.86	0.878	0.442	0.58	0.65	0.638	0.687	0.655	0.689	0.715
	wikiSERA-5	0.644	0.663	0.832	0.842	0.862	0.86	**0.911**	0.867	0.556	0.571	0.804	0.819	0.871	0.847	**0.891**	0.823	0.391	0.402	0.606	0.628	0.686	0.666	**0.739**	0.656
	wikiSERA-10	**0.799**	**0.817**	**0.876**	**0.884**	**0.898**	**0.892**	0.893	0.883	**0.757**	**0.797**	**0.86**	**0.867**	**0.888**	0.863	0.88	**0.882**	**0.568**	**0.598**	**0.679**	**0.687**	0.709	0.685	0.695	**0.702**
	wikiSERA-DIS-5	0.66	0.622	0.819	0.823	0.858	0.867	0.902	0.872	0.584	0.543	0.779	0.783	0.861	**0.871**	0.868	0.819	0.419	0.371	0.581	0.587	0.672	**0.701**	0.694	0.637
	wikiSERA-DIS-10	0.765	0.803	0.867	0.86	0.895	0.891	0.903	**0.891**	0.683	0.775	0.832	0.839	**0.888**	0.859	0.867	0.867	0.505	0.569	0.635	0.648	**0.714**	0.688	0.688	0.687

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.621	0.677	0.82	0.832	0.861	0.868	0.889	0.845	0.492	0.605	0.788	0.795	0.835	0.86	0.863	0.836	0.346	0.435	0.596	0.612	0.646	0.672	0.684	0.65
	SERA-10	**0.786**	0.803	0.871	**0.89**	0.886	**0.888**	0.885	0.874	**0.717**	0.774	0.858	**0.885**	**0.878**	0.866	0.877	0.872	**0.538**	0.581	0.681	**0.713**	**0.707**	0.692	0.709	0.694
	SERA-NP-5	0.659	0.746	0.752	0.857	0.888	0.847	0.813	0.808	0.584	0.671	0.735	0.831	0.874	0.834	0.787	0.773	0.407	0.487	0.538	0.651	0.702	0.663	0.602	0.595
	SERA-NP-10	0.766	**0.824**	**0.885**	0.867	0.891	0.886	0.892	0.853	0.688	**0.788**	**0.893**	0.832	0.849	**0.872**	**0.897**	0.844	0.503	**0.592**	**0.722**	0.652	0.662	**0.713**	**0.727**	0.659
	SERA-KW-5	0.627	0.681	0.839	0.843	0.859	0.859	0.888	0.858	0.505	0.61	0.822	0.797	0.837	0.854	0.863	0.813	0.355	0.438	0.625	0.608	0.653	0.668	0.689	0.641
	SERA-KW-10	0.776	0.809	0.873	0.885	0.886	0.887	0.888	0.871	0.69	0.779	0.854	0.873	0.869	0.869	0.882	0.88	0.514	**0.592**	0.675	0.7	0.692	0.686	0.714	0.711
	SERA-DIS-5	0.639	0.674	0.764	0.798	0.846	0.85	0.869	0.858	0.497	0.613	0.743	0.738	0.833	0.835	0.831	0.835	0.348	0.439	0.543	0.551	0.641	0.647	0.653	0.648
	SERA-DIS-10	0.754	0.794	0.844	0.856	0.877	0.872	0.885	**0.888**	0.642	0.761	0.833	0.831	0.877	0.833	0.863	**0.888**	0.456	0.562	0.65	0.645	0.69	0.643	0.69	**0.719**
	SERA-DIS-NP-5	0.641	0.737	0.739	0.835	0.857	0.845	0.818	0.78	0.575	0.676	0.74	0.822	0.835	0.841	0.797	0.739	0.39	0.491	0.533	0.624	0.631	0.653	0.591	0.557
	SERA-DIS-NP-10	0.716	0.817	0.849	0.862	**0.895**	0.882	**0.896**	0.831	0.647	0.777	0.861	0.846	0.874	**0.872**	0.893	0.827	0.464	0.567	0.672	0.658	0.692	0.71	0.708	0.644
	SERA-DIS-KW-5	0.642	0.678	0.78	0.812	0.837	0.84	0.867	0.866	0.52	0.612	0.752	0.767	0.807	0.834	0.831	0.824	0.369	0.441	0.55	0.58	0.617	0.644	0.653	0.648
	SERA-DIS-KW-10	0.747	0.802	0.852	0.856	0.875	0.871	0.889	0.884	0.627	0.78	0.836	0.815	0.877	0.839	0.857	0.875	0.442	0.578	0.648	0.636	0.683	0.653	0.686	0.713
	wikiSERA-5	0.645	0.664	0.832	0.842	0.861	0.859	**0.911**	0.867	0.556	0.572	0.803	0.816	0.868	0.844	**0.889**	0.821	0.391	0.402	0.603	0.626	0.684	0.664	**0.739**	0.654
	wikiSERA-10	**0.799**	**0.817**	**0.875**	**0.883**	**0.897**	**0.891**	0.892	0.882	**0.756**	**0.795**	**0.858**	**0.865**	0.884	0.859	0.878	**0.879**	**0.566**	**0.593**	**0.674**	**0.684**	0.707	0.683	0.693	**0.7**
	wikiSERA-DIS-5	0.661	0.623	0.819	0.823	0.857	0.867	0.901	0.872	0.585	0.545	0.779	0.782	0.858	**0.869**	0.866	0.817	0.422	0.371	0.581	0.587	0.667	**0.699**	0.691	0.637
	wikiSERA-DIS-10	0.766	0.803	0.867	0.86	0.894	**0.891**	0.902	**0.891**	0.683	0.774	0.83	0.837	**0.885**	0.856	0.866	0.864	0.502	0.567	0.632	0.648	**0.712**	0.685	0.685	0.684

Table A.3: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.663	0.738	0.764	0.863	0.868	**0.897**	0.88	0.845	0.559	0.691	0.725	0.833	0.867	0.866	**0.876**	0.822	0.402	0.505	0.541	0.643	0.69	0.695	**0.695**	0.634
	SERA-10	**0.81**	**0.812**	0.871	0.909	0.869	0.891	0.855	0.837	0.783	**0.793**	**0.845**	0.894	0.845	0.869	0.826	0.82	0.577	**0.594**	0.678	0.736	0.657	0.692	0.638	0.627
	SERA-NP-5	0.672	0.783	0.798	0.836	0.855	0.842	0.836	0.889	0.599	0.762	0.771	0.805	0.831	0.82	0.82	**0.894**	0.433	0.565	0.585	0.616	0.649	0.628	0.628	**0.729**
	SERA-NP-10	0.808	0.804	0.861	0.902	**0.902**	0.882	**0.883**	0.857	**0.805**	0.756	0.834	0.884	**0.885**	0.875	0.869	0.84	**0.612**	0.567	0.657	0.724	**0.722**	0.701	0.681	0.659
	SERA-KW-5	0.654	0.738	0.748	0.863	0.867	0.885	0.874	0.893	0.549	0.694	0.7	0.83	0.863	0.857	0.858	0.856	0.394	0.506	0.519	0.649	0.685	0.682	0.678	0.68
	SERA-KW-10	0.808	0.806	**0.872**	**0.913**	0.868	0.892	0.855	**0.899**	0.781	0.778	0.844	**0.9**	0.847	0.876	0.828	0.879	0.576	0.583	**0.68**	**0.749**	0.663	0.693	0.649	0.704
	SERA-DIS-5	0.633	0.721	0.777	0.87	0.835	0.887	0.865	0.847	0.526	0.663	0.747	0.822	0.813	0.87	0.856	0.825	0.385	0.48	0.556	0.629	0.615	0.696	0.673	0.638
	SERA-DIS-10	0.775	0.778	0.871	0.9	0.852	0.892	0.863	0.843	0.73	0.758	**0.845**	0.872	0.837	0.884	0.846	0.827	0.543	0.557	0.675	0.707	0.643	**0.706**	0.653	0.631
	SERA-DIS-NP-5	0.667	0.783	0.787	0.845	0.842	0.812	0.832	0.881	0.595	0.763	0.76	0.8	0.807	0.792	0.836	0.882	0.414	0.57	0.578	0.619	0.619	0.597	0.639	0.725
	SERA-DIS-NP-10	0.751	0.804	0.845	0.905	0.884	0.871	0.872	0.858	0.697	0.763	0.81	0.878	0.859	0.862	0.863	0.835	0.503	0.563	0.621	0.713	0.668	0.689	0.673	0.655
	SERA-DIS-KW-5	0.614	0.722	0.753	0.86	0.829	0.876	0.862	0.877	0.497	0.669	0.714	0.82	0.812	0.865	0.848	0.841	0.356	0.483	0.532	0.634	0.612	0.689	0.67	0.66
	SERA-DIS-KW-10	0.768	0.78	0.865	0.896	0.851	0.883	0.864	0.89	0.726	0.762	0.84	0.87	0.835	**0.885**	0.846	0.859	0.534	0.567	0.661	0.701	0.632	0.703	0.656	0.685
	wikiSERA-5	0.642	0.744	0.817	0.895	0.866	**0.895**	**0.891**	**0.876**	0.558	0.696	0.77	0.88	0.854	0.872	**0.869**	0.853	0.396	0.512	0.572	0.709	0.667	0.696	**0.693**	0.683
	wikiSERA-10	**0.785**	**0.824**	**0.859**	0.908	**0.879**	0.893	0.851	0.871	**0.754**	**0.806**	**0.816**	**0.891**	**0.861**	0.862	0.805	0.86	**0.545**	**0.613**	**0.641**	**0.739**	**0.682**	0.683	0.611	0.671
	wikiSERA-DIS-5	0.651	0.74	0.813	0.888	0.846	0.888	0.881	**0.876**	0.599	0.708	0.75	0.847	0.826	**0.886**	0.86	0.848	0.423	0.516	0.557	0.676	0.636	**0.705**	0.683	0.678
	wikiSERA-DIS-10	0.736	0.798	0.85	**0.917**	0.867	0.881	0.87	0.875	0.695	0.784	0.799	**0.891**	0.847	0.863	0.834	**0.871**	0.498	0.58	0.635	0.73	0.659	0.681	0.644	**0.688**

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.664	0.738	0.764	0.862	0.867	**0.896**	0.88	0.845	0.56	0.689	0.726	0.832	0.863	0.864	**0.875**	0.822	0.405	0.505	0.541	0.643	0.685	0.693	**0.695**	0.634
	SERA-10	**0.811**	**0.812**	0.87	0.908	0.869	0.891	0.855	0.837	0.781	**0.791**	0.844	0.892	0.843	0.867	0.824	0.818	0.575	**0.594**	0.678	0.734	0.654	0.692	0.638	0.627
	SERA-NP-5	0.673	0.783	0.798	0.836	0.854	0.84	0.836	0.889	0.599	0.759	0.771	0.804	0.827	0.817	0.818	**0.893**	0.435	0.563	0.585	0.619	0.644	0.626	0.625	**0.729**
	SERA-NP-10	0.809	0.804	0.861	0.901	**0.901**	0.881	**0.882**	0.857	**0.803**	0.753	0.833	0.882	**0.883**	0.871	0.868	0.838	**0.609**	0.567	0.657	0.721	**0.719**	0.698	0.679	0.659
	SERA-KW-5	0.655	0.738	0.748	0.862	0.866	0.884	0.873	0.892	0.55	0.692	0.701	0.828	0.859	0.854	0.857	0.854	0.396	0.506	0.52	0.649	0.68	0.679	0.678	0.68
	SERA-KW-10	0.808	0.806	**0.871**	**0.912**	0.868	0.891	0.855	**0.898**	0.778	0.776	0.843	**0.898**	0.844	0.874	0.826	0.876	0.574	0.583	**0.68**	**0.747**	0.658	0.693	0.649	0.701
	SERA-DIS-5	0.633	0.722	0.778	0.869	0.834	0.886	0.865	0.847	0.526	0.661	0.748	0.82	0.81	0.868	0.854	0.823	0.385	0.48	0.556	0.629	0.613	0.696	0.673	0.638
	SERA-DIS-10	0.775	0.778	0.87	0.9	0.851	0.891	0.863	0.842	0.729	0.756	**0.845**	0.87	0.833	0.882	0.843	0.825	0.543	0.557	0.675	0.705	0.638	**0.706**	0.653	0.631
	SERA-DIS-NP-5	0.668	0.782	0.786	0.845	0.841	0.811	0.831	0.881	0.596	0.761	0.759	0.799	0.802	0.789	0.834	0.882	0.417	0.568	0.578	0.619	0.614	0.595	0.637	0.723
	SERA-DIS-NP-10	0.752	0.803	0.844	0.905	0.883	0.871	0.872	0.858	0.696	0.76	0.809	0.876	0.855	0.859	0.86	0.833	0.503	0.563	0.621	0.71	0.666	0.687	0.671	0.653
	SERA-DIS-KW-5	0.614	0.722	0.754	0.859	0.828	0.875	0.862	0.877	0.497	0.668	0.714	0.818	0.809	0.863	0.846	0.84	0.356	0.483	0.532	0.632	0.609	0.687	0.67	0.66
	SERA-DIS-KW-10	0.769	0.779	0.865	0.896	0.85	0.883	0.864	0.889	0.725	0.759	0.839	0.867	0.831	**0.883**	0.844	0.856	0.534	0.567	0.661	0.699	0.63	0.703	0.656	0.683
	wikiSERA-5	0.643	0.744	0.817	0.894	0.865	**0.894**	**0.891**	**0.876**	0.559	0.694	0.771	0.878	0.85	0.871	**0.868**	0.852	0.396	0.512	0.575	0.709	0.662	0.693	**0.695**	0.681
	wikiSERA-10	**0.785**	**0.824**	**0.859**	0.908	**0.879**	0.892	0.85	0.871	**0.752**	**0.803**	**0.815**	**0.889**	**0.86**	0.861	0.803	0.857	**0.545**	**0.613**	**0.641**	**0.738**	**0.682**	0.683	0.611	0.671
	wikiSERA-DIS-5	0.652	0.74	0.814	0.888	0.845	0.887	0.881	**0.876**	0.599	0.706	0.751	0.845	0.822	**0.885**	0.859	0.847	0.423	0.516	0.56	0.673	0.633	**0.705**	0.683	0.676
	wikiSERA-DIS-10	0.736	0.798	0.849	**0.916**	0.866	0.881	0.87	0.875	0.694	0.781	0.799	**0.889**	0.844	0.862	0.832	**0.869**	0.498	0.58	0.635	0.728	0.656	0.681	0.644	**0.688**

Table A.4: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.605	0.707	0.769	0.896	0.801	**0.896**	**0.895**	0.872	0.485	0.674	0.761	0.868	0.778	0.855	0.872	0.872	0.333	0.487	0.556	0.698	0.593	0.671	0.695	0.693
	SERA-10	**0.795**	**0.808**	0.884	**0.903**	0.885	0.891	0.879	0.847	**0.745**	**0.783**	**0.863**	**0.894**	0.872	0.877	0.863	0.848	**0.561**	**0.574**	**0.683**	**0.718**	0.697	0.693	0.682	0.656
	SERA-NP-5	0.671	0.716	0.794	0.838	0.845	0.888	0.816	0.872	0.605	0.725	0.783	0.826	0.821	**0.883**	0.806	0.879	0.412	0.551	0.563	0.636	0.631	**0.698**	0.609	0.696
	SERA-NP-10	0.782	0.79	0.878	0.886	**0.91**	0.871	0.879	0.871	0.697	0.772	0.861	0.865	**0.906**	0.872	**0.89**	0.87	0.522	0.569	0.679	0.68	**0.736**	0.694	**0.717**	0.686
	SERA-KW-5	0.593	0.706	0.771	0.897	0.805	0.888	0.893	0.883	0.468	0.673	0.764	0.869	0.792	0.849	0.861	0.846	0.323	0.484	0.557	0.701	0.602	0.654	0.681	0.663
	SERA-KW-10	0.789	0.803	**0.885**	0.902	0.879	0.884	0.88	**0.897**	0.735	0.772	0.86	0.884	0.865	0.865	0.874	**0.886**	0.549	0.563	**0.683**	0.698	0.681	0.68	0.701	**0.718**
	SERA-DIS-5	0.602	0.691	0.784	0.894	0.766	0.863	0.886	0.887	0.496	0.687	0.757	0.875	0.735	0.834	0.856	**0.886**	0.343	0.48	0.547	0.698	0.543	0.644	0.671	0.709
	SERA-DIS-10	0.735	0.788	0.873	0.896	0.88	0.889	0.889	0.885	0.697	0.772	0.844	0.883	0.87	0.873	0.872	0.882	0.493	0.561	0.658	0.698	0.687	0.691	0.69	0.696
	SERA-DIS-NP-5	0.636	0.672	0.759	0.843	0.834	0.875	0.818	0.86	0.597	0.673	0.768	0.841	0.813	0.872	0.82	0.852	0.405	0.489	0.547	0.656	0.615	0.688	0.625	0.649
	SERA-DIS-NP-10	0.749	0.735	0.846	0.879	0.9	0.882	0.883	0.882	0.711	0.745	0.823	0.879	0.892	0.875	0.889	0.866	0.51	0.544	0.623	0.701	0.715	0.691	0.707	0.678
	SERA-DIS-KW-5	0.604	0.693	0.787	0.889	0.774	0.849	0.882	0.88	0.53	0.69	0.764	0.866	0.75	0.817	0.848	0.862	0.368	0.491	0.557	0.683	0.546	0.623	0.665	0.689
	SERA-DIS-KW-10	0.727	0.786	0.871	0.887	0.878	0.881	0.885	0.894	0.682	0.77	0.849	0.879	0.87	0.85	0.867	0.882	0.471	0.558	0.66	0.689	0.676	0.666	0.69	0.707
	wikiSERA-5	0.605	0.682	0.771	0.86	0.849	0.893	0.908	**0.918**	0.501	0.648	0.75	0.829	0.839	**0.883**	0.87	**0.921**	0.349	0.459	0.554	0.639	0.653	**0.701**	0.702	**0.758**
	wikiSERA-10	**0.785**	**0.778**	**0.879**	**0.911**	**0.885**	0.892	0.89	0.881	0.717	**0.739**	**0.843**	0.91	0.867	0.868	0.878	0.892	0.523	**0.546**	**0.672**	**0.74**	0.687	0.689	0.698	0.707
	wikiSERA-DIS-5	0.639	0.665	0.754	0.864	0.818	0.878	0.903	0.914	0.599	0.651	0.724	0.858	0.799	0.869	0.874	0.898	0.42	0.452	0.528	0.661	0.602	0.678	0.684	0.73
	wikiSERA-DIS-10	0.756	0.741	0.85	0.902	0.881	**0.895**	**0.913**	0.917	**0.738**	0.702	0.796	**0.911**	**0.882**	**0.883**	**0.897**	0.91	**0.545**	0.504	0.61	0.734	**0.701**	0.694	**0.73**	0.739

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.606	0.708	0.769	0.896	0.799	**0.895**	**0.895**	0.872	0.483	0.673	0.761	0.868	0.775	0.852	0.869	0.869	0.333	0.487	0.556	0.698	0.589	0.669	0.693	0.691
	SERA-10	**0.795**	**0.807**	0.884	**0.903**	0.885	0.891	0.879	0.847	**0.744**	**0.781**	**0.863**	**0.892**	0.871	0.874	0.861	0.847	**0.561**	**0.571**	**0.682**	**0.717**	0.697	0.691	0.682	0.656
	SERA-NP-5	0.671	0.716	0.794	0.838	0.844	0.887	0.816	0.872	0.604	0.723	0.781	0.826	0.819	**0.882**	0.804	0.879	0.412	0.548	0.563	0.636	0.629	**0.698**	0.607	0.696
	SERA-NP-10	0.783	0.79	0.879	0.885	**0.909**	0.871	0.879	0.87	0.697	0.77	0.86	0.864	**0.905**	0.869	**0.888**	0.867	0.525	0.566	0.679	0.68	**0.733**	0.691	**0.714**	0.683
	SERA-KW-5	0.594	0.706	0.771	0.897	0.804	0.887	0.892	0.883	0.467	0.674	0.764	0.869	0.789	0.845	0.857	0.844	0.323	0.484	0.557	0.701	0.6	0.652	0.679	0.663
	SERA-KW-10	0.789	0.803	**0.885**	0.901	0.878	0.884	0.88	**0.897**	0.734	0.77	0.86	0.882	0.864	0.862	0.871	**0.885**	0.55	0.56	0.68	0.698	0.681	0.677	0.701	**0.716**
	SERA-DIS-5	0.603	0.691	0.784	0.893	0.765	0.862	0.885	0.886	0.494	0.686	0.757	0.874	0.733	0.831	0.853	0.883	0.343	0.48	0.547	0.696	0.54	0.642	0.668	0.707
	SERA-DIS-10	0.735	0.788	0.873	0.895	0.879	0.889	0.889	0.884	0.695	0.769	0.843	0.881	0.869	0.871	0.87	0.88	0.493	0.558	0.658	0.695	0.687	0.689	0.69	0.696
	SERA-DIS-NP-5	0.637	0.672	0.759	0.843	0.833	0.875	0.818	0.86	0.597	0.672	0.766	0.84	0.81	0.869	0.817	0.852	0.407	0.489	0.547	0.656	0.613	0.685	0.623	0.649
	SERA-DIS-NP-10	0.75	0.735	0.846	0.879	0.899	0.881	0.882	0.882	0.712	0.744	0.822	0.876	0.889	0.871	0.887	0.864	0.512	0.541	0.623	0.701	0.71	0.689	0.707	0.676
	SERA-DIS-KW-5	0.604	0.693	0.787	0.889	0.773	0.848	0.882	0.88	0.528	0.689	0.764	0.866	0.748	0.814	0.845	0.861	0.368	0.491	0.557	0.683	0.544	0.62	0.662	0.689
	SERA-DIS-KW-10	0.728	0.786	0.871	0.886	0.877	0.88	0.885	0.894	0.681	0.768	0.849	0.876	0.869	0.848	0.864	0.882	0.471	0.558	0.66	0.689	0.676	0.664	0.69	0.705
	wikiSERA-5	0.606	0.682	0.771	0.86	0.848	0.892	0.908	**0.917**	0.5	0.647	0.749	0.827	0.835	**0.881**	0.869	**0.918**	0.349	0.457	0.554	0.639	0.648	**0.699**	0.702	**0.755**
	wikiSERA-10	**0.786**	**0.778**	**0.879**	**0.911**	**0.885**	0.891	0.89	0.88	0.716	**0.737**	**0.843**	0.908	0.865	0.865	0.875	0.89	0.523	**0.544**	**0.672**	**0.74**	0.687	0.687	0.698	0.704
	wikiSERA-DIS-5	0.64	0.665	0.754	0.864	0.816	0.877	0.902	0.913	0.597	0.65	0.724	0.855	0.795	0.867	0.872	0.896	0.42	0.452	0.528	0.661	0.597	0.676	0.681	0.73
	wikiSERA-DIS-10	0.756	0.741	0.85	0.902	0.88	**0.894**	**0.912**	**0.917**	**0.735**	0.699	0.795	**0.91**	**0.88**	0.88	**0.895**	0.907	**0.542**	0.504	0.61	0.734	**0.699**	0.691	**0.73**	0.736

Table A.5: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_3$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.638	0.771	0.821	**0.9**	0.867	0.883	0.885	0.844	0.526	0.711	0.8	**0.869**	0.854	0.852	0.884	0.854	0.367	0.528	0.604	**0.701**	0.667	0.667	0.696	0.667
	SERA-10	0.818	0.823	**0.905**	0.867	**0.894**	**0.9**	0.872	0.877	0.76	0.793	**0.874**	0.859	0.881	0.885	0.853	0.887	0.574	0.595	**0.702**	0.679	0.702	0.704	0.668	0.704
	SERA-NP-5	0.659	0.737	0.772	0.859	0.835	0.873	0.858	0.885	0.585	0.666	0.776	0.827	0.799	0.835	0.817	0.874	0.408	0.486	0.564	0.639	0.612	0.662	0.639	0.69
	SERA-NP-10	**0.831**	**0.845**	0.879	0.879	0.889	0.89	**0.906**	0.886	**0.803**	**0.817**	0.847	0.866	**0.891**	0.864	0.881	0.878	**0.596**	**0.624**	0.664	0.686	**0.715**	0.686	**0.713**	0.702
	SERA-KW-5	0.636	0.775	0.821	0.895	0.867	0.893	0.876	0.861	0.535	0.72	0.788	0.86	0.855	0.86	0.873	0.824	0.375	0.535	0.584	0.694	0.671	0.673	0.681	0.65
	SERA-KW-10	0.815	0.825	**0.905**	0.866	0.889	0.899	0.878	0.909	0.758	0.795	0.872	0.853	0.881	0.883	0.865	0.887	0.561	0.595	**0.702**	0.674	0.7	0.707	0.68	**0.727**
	SERA-DIS-5	0.586	0.747	0.79	0.881	0.865	0.853	0.88	0.864	0.443	0.705	0.772	0.854	0.866	0.815	0.861	0.86	0.31	0.504	0.572	0.679	0.672	0.621	0.672	0.671
	SERA-DIS-10	0.776	0.818	0.868	0.857	**0.894**	0.893	0.893	0.888	0.713	0.797	0.841	0.839	0.889	0.874	**0.889**	**0.892**	0.517	0.585	0.655	0.664	0.7	0.701	**0.713**	0.706
	SERA-DIS-NP-5	0.655	0.71	0.729	0.839	0.872	0.83	0.841	0.877	0.56	0.632	0.73	0.818	0.862	0.781	0.816	0.828	0.377	0.465	0.508	0.632	0.688	0.589	0.626	0.649
	SERA-DIS-NP-10	0.758	0.811	0.84	0.861	0.886	0.878	0.88	0.896	0.715	0.761	0.84	0.838	0.884	0.867	0.852	0.859	0.514	0.56	0.649	0.648	0.696	0.685	0.673	0.679
	SERA-DIS-KW-5	0.559	0.761	0.797	0.875	0.863	0.867	0.876	0.873	0.404	0.724	0.768	0.847	0.866	0.829	0.847	0.839	0.278	0.527	0.562	0.665	0.668	0.639	0.661	0.664
	SERA-DIS-KW-10	0.767	0.829	0.871	0.849	**0.894**	**0.9**	0.896	**0.913**	0.707	0.806	0.845	0.833	0.89	**0.892**	0.885	0.882	0.516	0.597	0.655	0.653	0.704	**0.713**	0.711	0.714
	wikiSERA-5	0.677	0.775	0.831	**0.884**	0.88	0.891	0.903	0.873	0.564	0.755	0.817	0.844	0.848	**0.899**	0.896	0.861	0.401	0.562	0.628	0.67	0.666	**0.726**	0.724	**0.689**
	wikiSERA-10	**0.794**	**0.836**	**0.892**	0.866	**0.893**	**0.899**	0.903	**0.888**	**0.748**	0.81	**0.846**	**0.852**	**0.88**	0.876	**0.904**	0.864	**0.551**	0.608	**0.678**	**0.68**	**0.706**	0.701	**0.734**	0.684
	wikiSERA-DIS-5	0.648	0.775	0.802	0.853	0.864	0.859	0.897	0.853	0.514	0.762	0.79	0.82	0.845	0.852	0.846	0.817	0.36	0.563	0.584	0.641	0.662	0.656	0.67	0.629
	wikiSERA-DIS-10	0.762	0.832	0.861	0.863	0.889	0.895	**0.914**	0.884	0.698	**0.812**	0.825	0.842	0.866	0.89	0.886	**0.867**	0.511	**0.614**	0.639	0.666	0.694	0.716	0.71	0.685

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.638	0.772	0.821	**0.899**	0.866	0.882	0.884	0.843	0.526	0.712	0.8	**0.867**	0.851	0.848	0.882	0.851	0.367	0.528	0.604	**0.699**	0.664	0.664	0.694	0.667
	SERA-10	0.818	0.823	0.904	0.867	0.893	0.899	0.871	0.877	0.756	0.792	**0.871**	0.855	0.878	0.881	0.851	0.886	0.572	0.595	**0.698**	0.675	0.701	0.702	0.666	0.701
	SERA-NP-5	0.659	0.737	0.771	0.858	0.834	0.872	0.858	0.884	0.584	0.666	0.773	0.825	0.797	0.832	0.814	0.871	0.408	0.486	0.559	0.636	0.612	0.66	0.637	0.688
	SERA-NP-10	**0.831**	**0.845**	0.877	0.878	0.889	0.89	**0.905**	0.885	**0.801**	**0.817**	0.843	0.862	**0.889**	0.86	0.879	0.876	**0.596**	**0.624**	0.659	0.684	**0.713**	0.683	**0.711**	0.702
	SERA-KW-5	0.637	0.776	0.821	0.895	0.866	0.893	0.875	0.861	0.535	0.72	0.788	0.858	0.852	0.856	0.87	0.822	0.375	0.535	0.584	0.692	0.669	0.671	0.678	0.65
	SERA-KW-10	0.815	0.825	**0.905**	0.866	0.888	0.899	0.877	0.908	0.754	0.794	0.868	0.849	0.878	0.879	0.863	0.883	0.558	0.595	**0.698**	0.672	0.697	0.704	0.677	**0.724**
	SERA-DIS-5	0.587	0.748	0.789	0.881	0.864	0.852	0.879	0.863	0.442	0.705	0.772	0.853	0.864	0.812	0.858	0.856	0.31	0.504	0.572	0.677	0.672	0.619	0.667	0.668
	SERA-DIS-10	0.777	0.818	0.867	0.856	**0.894**	0.892	0.892	0.888	0.709	0.796	0.838	0.835	0.886	0.871	**0.886**	**0.89**	0.515	0.585	0.65	0.661	0.695	0.699	**0.711**	0.704
	SERA-DIS-NP-5	0.655	0.711	0.728	0.838	0.871	0.829	0.841	0.877	0.561	0.631	0.727	0.816	0.861	0.78	0.814	0.826	0.377	0.463	0.503	0.632	0.688	0.589	0.624	0.649
	SERA-DIS-NP-10	0.759	0.811	0.839	0.86	0.885	0.877	0.879	0.896	0.715	0.759	0.836	0.834	0.881	0.864	0.85	0.857	0.512	0.56	0.644	0.643	0.691	0.683	0.671	0.679
	SERA-DIS-KW-5	0.56	0.761	0.796	0.875	0.862	0.867	0.875	0.873	0.402	0.724	0.768	0.846	0.864	0.825	0.843	0.835	0.275	0.527	0.562	0.665	0.668	0.637	0.656	0.659
	SERA-DIS-KW-10	0.767	0.829	0.87	0.849	0.893	**0.9**	0.895	**0.912**	0.703	0.805	0.842	0.829	0.887	**0.888**	0.883	0.877	0.514	0.597	0.65	0.65	0.701	**0.711**	0.708	0.71
	wikiSERA-5	0.678	0.776	0.831	**0.884**	0.879	0.89	0.902	0.872	0.563	0.757	0.817	0.843	0.846	**0.896**	0.893	0.859	0.401	0.562	0.628	0.67	0.661	**0.724**	0.719	**0.689**
	wikiSERA-10	**0.794**	**0.836**	**0.892**	0.865	**0.892**	**0.898**	0.903	**0.887**	**0.742**	0.809	**0.843**	**0.848**	**0.878**	0.872	**0.902**	0.861	**0.549**	0.608	**0.673**	**0.678**	**0.704**	0.698	**0.734**	0.682
	wikiSERA-DIS-5	0.649	0.776	0.801	0.853	0.864	0.858	0.896	0.852	0.512	0.764	0.79	0.819	0.842	0.851	0.843	0.813	0.36	0.56	0.584	0.638	0.66	0.656	0.667	0.624
	wikiSERA-DIS-10	0.763	0.832	0.86	0.862	0.889	0.895	**0.914**	0.883	0.694	**0.811**	0.822	0.839	0.863	0.887	0.885	**0.863**	0.509	**0.612**	0.637	0.664	0.691	0.713	0.71	0.683

Table A.6: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_4$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.646	0.745	0.809	0.897	0.873	**0.911**	**0.91**	0.877	0.543	0.701	0.804	0.881	0.868	0.887	0.89	0.872	0.379	0.505	0.61	0.709	0.684	0.708	0.71	0.699
	SERA-10	**0.815**	0.831	0.891	**0.913**	0.897	0.905	0.888	0.872	**0.77**	0.826	0.867	0.898	0.889	0.886	0.874	0.865	**0.574**	0.626	0.693	0.73	0.707	0.698	0.699	0.677
	SERA-NP-5	0.686	0.795	0.815	0.876	0.892	0.893	0.851	0.885	0.635	0.787	0.823	0.869	0.882	0.877	0.847	0.888	0.45	0.592	0.623	0.691	0.696	0.707	0.653	0.707
	SERA-NP-10	0.803	**0.837**	**0.897**	0.906	**0.919**	0.893	0.903	0.879	0.743	0.823	**0.888**	0.9	**0.907**	0.885	0.903	0.871	0.538	0.618	**0.716**	0.727	**0.736**	0.714	0.734	0.683
	SERA-KW-5	0.643	0.744	0.811	0.901	0.872	0.904	0.908	0.898	0.548	0.706	0.802	0.891	0.867	0.881	0.892	0.87	0.39	0.518	0.603	0.718	0.687	0.699	0.71	0.704
	SERA-KW-10	0.809	0.83	0.892	**0.913**	0.895	0.903	0.89	0.901	0.765	**0.833**	0.868	0.895	0.888	0.887	0.876	**0.9**	0.564	**0.637**	0.691	0.733	0.705	0.701	0.704	**0.733**
	SERA-DIS-5	0.645	0.74	0.813	0.897	0.853	0.893	0.9	0.893	0.545	0.71	0.795	0.881	0.848	0.883	0.874	0.886	0.387	0.522	0.598	0.707	0.652	0.712	0.699	0.711
	SERA-DIS-10	0.773	0.816	0.885	0.905	0.89	0.904	0.898	0.894	0.733	0.813	0.865	0.892	0.887	**0.895**	0.886	0.893	0.54	0.614	0.676	0.735	0.701	0.713	0.71	0.721
	SERA-DIS-NP-5	0.67	0.78	0.805	0.885	0.883	0.884	0.863	0.891	0.606	0.789	0.825	0.876	0.865	0.877	0.856	0.89	0.416	0.589	0.621	0.698	0.676	0.707	0.67	0.713
	SERA-DIS-NP-10	0.757	0.821	0.873	0.912	0.916	0.896	0.907	0.883	0.706	0.819	0.871	**0.906**	0.894	0.888	**0.916**	0.877	0.51	0.617	0.688	**0.736**	0.71	**0.718**	**0.745**	0.691
	SERA-DIS-KW-5	0.64	0.742	0.812	0.897	0.85	0.884	0.899	0.899	0.557	0.722	0.802	0.876	0.842	0.874	0.874	0.87	0.395	0.532	0.604	0.699	0.639	0.694	0.698	0.699
	SERA-DIS-KW-10	0.768	0.819	0.884	0.902	0.887	0.898	0.898	**0.905**	0.732	0.824	0.867	0.887	0.879	0.891	0.887	0.889	0.534	0.619	0.677	0.724	0.69	0.708	0.713	0.719
	wikiSERA-5	0.651	0.737	0.836	0.9	0.884	**0.912**	**0.926**	0.911	0.598	0.705	0.808	0.885	0.883	0.903	**0.914**	0.903	0.423	0.512	0.618	0.701	0.7	0.723	**0.749**	**0.735**
	wikiSERA-10	**0.807**	**0.833**	**0.891**	0.917	**0.903**	0.909	0.895	0.895	**0.778**	**0.83**	**0.862**	0.911	0.887	0.893	0.88	0.9	**0.573**	**0.632**	**0.683**	**0.748**	0.709	0.713	0.699	0.721
	wikiSERA-DIS-5	0.67	0.726	0.835	0.904	0.875	0.909	0.923	**0.918**	0.63	0.706	0.808	0.897	0.879	**0.907**	**0.914**	0.903	0.451	0.509	0.617	0.72	0.691	**0.739**	0.748	0.734
	wikiSERA-DIS-10	0.774	0.813	0.885	**0.92**	0.9	0.909	0.914	0.915	0.749	0.806	0.847	**0.916**	**0.893**	0.898	0.889	**0.909**	0.549	0.607	0.668	**0.748**	**0.713**	0.722	0.718	0.731

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.647	0.745	0.809	0.897	0.871	**0.91**	**0.909**	0.877	0.544	0.701	0.805	0.88	0.864	0.884	0.887	0.871	0.379	0.505	0.61	0.709	0.679	0.706	0.71	0.699
	SERA-10	**0.815**	0.831	0.891	**0.913**	0.897	0.905	0.887	0.871	**0.768**	0.823	0.867	0.895	0.886	0.883	0.871	0.863	**0.572**	0.624	0.693	0.728	0.705	0.695	0.699	0.677
	SERA-NP-5	0.687	0.795	0.814	0.876	0.891	0.892	0.851	0.885	0.634	0.785	0.821	0.868	0.879	0.875	0.845	0.887	0.45	0.589	0.621	0.691	0.693	0.704	0.65	0.707
	SERA-NP-10	0.803	**0.837**	**0.897**	0.906	**0.919**	0.892	0.903	0.879	0.742	0.821	**0.887**	0.897	**0.906**	0.881	0.902	0.869	0.538	0.618	**0.716**	0.724	**0.734**	0.712	0.731	0.683
	SERA-KW-5	0.644	0.744	0.811	0.901	0.87	0.903	0.907	0.897	0.548	0.706	0.802	0.89	0.863	0.878	0.889	0.868	0.39	0.518	0.603	0.718	0.682	0.696	0.71	0.704
	SERA-KW-10	0.81	0.83	0.892	**0.913**	0.894	0.903	0.89	0.9	0.763	**0.83**	0.867	0.892	0.885	0.884	0.873	**0.897**	0.564	**0.634**	0.691	0.73	0.702	0.699	0.704	**0.73**
	SERA-DIS-5	0.645	0.74	0.813	0.897	0.851	0.892	0.9	0.893	0.545	0.708	0.795	0.88	0.845	0.881	0.871	0.884	0.389	0.522	0.598	0.707	0.649	0.71	0.696	0.711
	SERA-DIS-10	0.773	0.816	0.885	0.905	0.889	0.903	0.897	0.894	0.731	0.809	0.864	0.889	0.884	**0.892**	0.882	0.891	0.54	0.612	0.676	0.733	0.699	0.711	0.707	0.721
	SERA-DIS-NP-5	0.671	0.78	0.805	0.885	0.882	0.884	0.863	0.89	0.607	0.787	0.824	0.875	0.861	0.874	0.854	0.887	0.418	0.586	0.619	0.698	0.671	0.705	0.667	0.711
	SERA-DIS-NP-10	0.758	0.821	0.873	0.911	0.916	0.895	0.907	0.882	0.706	0.817	0.87	**0.903**	0.891	0.884	**0.914**	0.874	0.512	0.617	0.688	**0.734**	0.707	**0.716**	**0.742**	0.689
	SERA-DIS-KW-5	0.64	0.742	0.812	0.897	0.848	0.883	0.898	0.898	0.557	0.721	0.802	0.874	0.839	0.872	0.872	0.868	0.397	0.532	0.602	0.699	0.637	0.691	0.698	0.699
	SERA-DIS-KW-10	0.768	0.819	0.884	0.902	0.887	0.897	0.898	**0.904**	0.731	0.821	0.867	0.884	0.876	0.888	0.884	0.886	0.534	0.616	0.677	0.724	0.688	0.706	0.711	0.717
	wikiSERA-5	0.652	0.738	0.836	0.9	0.883	**0.911**	**0.926**	0.911	0.598	0.704	0.808	0.883	0.879	0.901	**0.914**	0.901	0.426	0.51	0.618	0.701	0.695	0.721	**0.749**	0.732
	wikiSERA-10	**0.808**	**0.833**	**0.891**	0.917	**0.903**	0.908	0.895	0.894	**0.776**	**0.827**	**0.86**	0.909	0.885	0.89	0.878	0.897	**0.573**	**0.63**	**0.683**	**0.748**	0.707	0.711	0.699	0.719
	wikiSERA-DIS-5	0.671	0.726	0.836	0.904	0.874	0.909	0.923	**0.918**	0.629	0.705	0.808	0.895	0.875	**0.905**	0.912	0.901	0.453	0.506	0.617	0.72	0.687	**0.736**	0.748	**0.734**
	wikiSERA-DIS-10	0.775	0.813	0.885	**0.92**	0.899	0.909	0.914	0.914	0.747	0.803	0.846	**0.914**	**0.89**	0.896	0.887	**0.907**	0.546	0.604	0.668	**0.748**	**0.711**	0.719	0.716	0.729

Table A.7: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.656	0.764	0.827	0.897	0.891	**0.906**	0.909	0.868	0.539	0.709	0.823	0.868	0.887	0.883	**0.908**	0.862	0.384	0.517	0.631	0.694	0.707	0.706	0.729	0.682
	SERA-10	**0.825**	0.832	0.898	0.902	0.899	**0.906**	0.886	0.881	0.795	0.818	0.881	0.89	0.882	0.891	0.869	0.877	**0.592**	0.614	0.711	0.722	0.701	**0.71**	0.687	0.691
	SERA-NP-5	0.685	0.808	0.81	0.882	0.887	0.88	0.869	0.887	0.619	0.777	0.813	0.874	0.867	0.857	0.854	0.878	0.426	0.58	0.622	0.692	0.682	0.691	0.675	0.702
	SERA-NP-10	0.821	**0.85**	0.898	0.902	**0.913**	0.897	**0.91**	0.884	0.79	**0.825**	**0.895**	0.883	**0.901**	0.876	0.907	0.88	0.579	**0.625**	**0.718**	0.7	**0.731**	0.704	**0.738**	0.705
	SERA-KW-5	0.655	0.766	0.828	0.9	0.887	0.905	0.904	0.887	0.534	0.731	0.814	0.877	0.884	0.88	0.894	0.849	0.38	0.535	0.619	0.702	0.715	0.698	0.715	0.677
	SERA-KW-10	0.82	0.833	**0.9**	0.902	0.896	**0.906**	0.89	0.907	**0.798**	0.809	0.883	0.889	0.879	**0.892**	0.876	0.892	**0.592**	0.604	0.707	**0.724**	0.698	0.709	0.696	0.728
	SERA-DIS-5	0.638	0.757	0.811	0.89	0.881	0.89	0.902	0.885	0.517	0.722	0.809	0.857	0.876	0.869	0.89	0.874	0.362	0.526	0.616	0.681	0.694	0.689	0.716	0.694
	SERA-DIS-10	0.787	0.822	0.882	0.892	0.893	0.902	0.899	0.896	0.748	0.814	0.864	0.874	0.887	0.884	0.882	**0.895**	0.551	0.614	0.679	0.71	0.701	0.701	0.71	0.718
	SERA-DIS-NP-5	0.677	0.796	0.793	0.882	0.892	0.862	0.874	0.889	0.6	0.767	0.803	0.86	0.881	0.832	0.869	0.871	0.415	0.569	0.602	0.683	0.71	0.655	0.687	0.707
	SERA-DIS-NP-10	0.762	0.839	0.871	**0.906**	0.911	0.894	0.906	0.889	0.72	0.807	0.875	**0.892**	0.898	0.879	0.896	0.88	0.516	0.611	0.689	0.715	0.724	**0.71**	0.721	0.709
	SERA-DIS-KW-5	0.624	0.763	0.811	0.889	0.874	0.891	0.9	0.896	0.487	0.728	0.806	0.865	0.87	0.882	0.88	0.863	0.349	0.528	0.61	0.698	0.682	0.701	0.706	0.7
	SERA-DIS-KW-10	0.782	0.83	0.884	0.89	0.89	0.902	0.902	**0.911**	0.735	0.819	0.867	0.872	0.881	0.891	0.883	0.892	0.538	0.616	0.679	0.708	0.692	**0.71**	0.711	**0.735**
	wikiSERA-5	0.677	0.77	0.859	0.902	0.899	**0.912**	**0.926**	0.897	0.59	0.738	0.84	0.883	0.891	0.908	**0.923**	0.886	0.413	0.537	0.647	0.706	0.707	0.727	**0.765**	**0.708**
	wikiSERA-10	**0.812**	**0.847**	**0.893**	0.902	**0.906**	0.911	0.9	0.895	**0.79**	**0.842**	0.869	0.883	0.888	0.889	0.882	**0.889**	**0.585**	**0.641**	0.69	0.718	**0.719**	0.713	0.698	0.706
	wikiSERA-DIS-5	0.677	0.762	0.846	0.897	0.893	0.903	0.919	0.896	0.601	0.725	0.82	0.879	**0.892**	**0.909**	0.897	0.876	0.418	0.522	0.619	0.7	0.71	**0.746**	0.729	0.705
	wikiSERA-DIS-10	0.779	0.84	0.887	**0.91**	0.904	0.906	0.914	**0.901**	0.751	0.836	**0.875**	**0.905**	0.889	0.896	0.887	0.888	0.547	0.623	**0.701**	**0.745**	0.716	0.727	0.717	0.705

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.656	0.765	0.827	0.896	0.89	**0.906**	0.908	0.868	0.54	0.709	0.824	0.866	0.882	0.88	**0.905**	0.861	0.386	0.517	0.629	0.694	0.702	0.704	0.729	0.682
	SERA-10	**0.825**	0.832	0.897	0.901	0.899	**0.906**	0.885	0.881	0.792	0.815	0.879	0.886	0.879	0.887	0.867	0.875	**0.589**	0.614	0.711	0.717	0.699	0.707	0.685	0.689
	SERA-NP-5	0.686	0.809	0.81	0.881	0.886	0.879	0.868	0.887	0.619	0.775	0.812	0.874	0.864	0.853	0.852	0.876	0.426	0.58	0.62	0.692	0.677	0.689	0.673	0.702
	SERA-NP-10	0.821	**0.85**	0.898	0.901	**0.912**	0.897	**0.91**	0.883	0.789	**0.824**	**0.892**	0.88	**0.899**	0.872	**0.905**	0.878	0.579	**0.625**	**0.715**	0.697	**0.729**	0.701	**0.736**	0.705
	SERA-KW-5	0.656	0.767	0.828	0.9	0.886	0.904	0.904	0.887	0.535	0.731	0.815	0.875	0.88	0.876	0.892	0.847	0.382	0.535	0.616	0.702	0.71	0.695	0.715	0.677
	SERA-KW-10	0.821	0.833	**0.899**	0.901	0.896	0.905	0.889	0.906	**0.796**	0.807	0.881	0.885	0.876	**0.888**	0.874	0.888	**0.589**	0.604	0.705	**0.719**	0.695	0.707	0.693	0.725
	SERA-DIS-5	0.638	0.757	0.811	0.89	0.88	0.89	0.901	0.884	0.518	0.722	0.81	0.856	0.872	0.867	0.887	0.872	0.362	0.526	0.614	0.681	0.689	0.686	0.713	0.694
	SERA-DIS-10	0.787	0.822	0.881	0.891	0.893	0.902	0.898	0.895	0.746	0.812	0.863	0.871	0.883	0.881	0.879	**0.892**	0.551	0.614	0.677	0.71	0.696	0.699	0.707	0.716
	SERA-DIS-NP-5	0.677	0.797	0.792	0.882	0.891	0.862	0.874	0.889	0.6	0.766	0.801	0.859	0.878	0.829	0.867	0.869	0.415	0.569	0.6	0.683	0.707	0.653	0.684	0.705
	SERA-DIS-NP-10	0.763	0.838	0.871	**0.905**	0.91	0.893	0.906	0.889	0.719	0.805	0.873	**0.888**	0.895	0.875	0.894	0.877	0.516	0.611	0.687	0.713	0.722	**0.708**	0.718	0.706
	SERA-DIS-KW-5	0.625	0.764	0.811	0.889	0.873	0.89	0.899	0.896	0.488	0.729	0.806	0.864	0.867	0.879	0.877	0.86	0.349	0.528	0.608	0.698	0.677	0.699	0.704	0.7
	SERA-DIS-KW-10	0.782	0.829	0.884	0.889	0.889	0.901	0.901	**0.91**	0.733	0.817	0.866	0.869	0.878	**0.888**	0.881	0.888	0.538	0.616	0.677	0.706	0.687	0.707	0.708	**0.73**
	wikiSERA-5	0.678	0.771	0.859	0.901	0.898	**0.911**	**0.926**	0.897	0.59	0.738	0.84	0.881	0.888	0.906	**0.922**	0.884	0.413	0.537	0.645	0.706	0.705	0.725	**0.765**	**0.706**
	wikiSERA-10	**0.812**	**0.847**	**0.893**	0.901	**0.905**	0.91	0.9	0.895	**0.786**	**0.84**	0.866	0.88	0.885	0.885	0.881	**0.886**	**0.583**	**0.641**	0.687	0.716	**0.717**	0.711	0.698	0.703
	wikiSERA-DIS-5	0.678	0.763	0.846	0.897	0.892	0.903	0.919	0.895	0.601	0.725	0.821	0.878	**0.889**	**0.907**	0.895	0.874	0.418	0.522	0.616	0.7	0.708	**0.743**	0.727	0.702
	wikiSERA-DIS-10	0.78	0.84	0.887	**0.909**	0.904	0.905	0.913	**0.9**	0.749	0.833	**0.873**	**0.902**	0.886	0.893	0.885	0.885	0.545	0.623	**0.699**	**0.742**	0.713	0.724	0.714	0.702

Table A.8: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.651	0.768	0.812	**0.913**	0.87	**0.913**	**0.906**	0.874	0.552	0.733	0.81	0.895	0.875	0.885	**0.906**	0.875	0.388	0.537	0.618	0.724	0.689	0.706	0.726	0.696
	SERA-10	0.827	0.831	0.9	0.906	0.898	0.906	0.88	0.869	0.795	0.809	0.88	0.89	0.88	0.885	0.867	0.87	**0.595**	0.606	0.71	0.724	0.688	0.697	0.687	0.671
	SERA-NP-5	0.688	0.79	0.815	0.87	0.873	0.894	0.863	0.906	0.633	0.758	0.807	0.864	0.871	0.87	0.857	**0.913**	0.433	0.564	0.603	0.687	0.683	0.698	0.673	**0.736**
	SERA-NP-10	**0.828**	**0.836**	0.893	0.906	**0.915**	0.892	0.902	0.886	0.79	0.815	**0.883**	0.894	**0.918**	0.874	0.899	0.882	0.58	**0.627**	0.701	0.719	**0.759**	0.693	**0.727**	0.701
	SERA-KW-5	0.645	0.767	0.806	0.912	0.869	0.911	0.9	0.895	0.551	0.737	0.8	0.888	0.871	0.881	0.89	0.861	0.39	0.539	0.606	0.721	0.69	0.695	0.707	0.685
	SERA-KW-10	0.823	0.829	**0.901**	0.906	0.894	0.903	0.883	**0.913**	**0.801**	0.815	0.879	0.889	0.878	0.886	0.876	0.897	0.593	0.609	**0.711**	0.718	0.687	0.696	0.695	0.728
	SERA-DIS-5	0.625	0.75	0.821	0.909	0.853	0.891	0.899	0.893	0.532	0.727	0.813	0.884	0.854	0.867	0.888	0.888	0.368	0.53	0.615	0.716	0.656	0.677	0.713	0.722
	SERA-DIS-10	0.778	0.814	0.887	0.901	0.893	0.905	0.895	0.891	0.748	0.817	0.867	0.884	0.887	**0.894**	0.886	0.891	0.541	0.609	0.682	0.719	0.701	**0.714**	0.706	0.713
	SERA-DIS-NP-5	0.677	0.774	0.79	0.872	0.878	0.872	0.865	0.909	0.624	0.749	0.799	0.857	0.861	0.853	0.863	0.894	0.429	0.551	0.591	0.689	0.676	0.675	0.678	0.711
	SERA-DIS-NP-10	0.774	0.811	0.867	0.903	0.909	0.892	0.896	0.9	0.748	0.79	0.863	**0.899**	0.903	0.885	0.89	0.881	0.539	0.59	0.676	**0.727**	0.733	0.711	0.723	0.701
	SERA-DIS-KW-5	0.61	0.754	0.817	0.904	0.851	0.889	0.896	0.897	0.506	0.739	0.81	0.882	0.854	0.872	0.879	0.864	0.356	0.543	0.607	0.71	0.66	0.677	0.701	0.691
	SERA-DIS-KW-10	0.773	0.818	0.887	0.894	0.891	0.901	0.896	0.912	0.737	**0.818**	0.87	0.881	0.884	0.892	0.887	0.889	0.533	0.612	0.677	0.708	0.695	0.711	0.709	0.721
	wikiSERA-5	0.666	0.765	0.837	0.905	0.887	**0.918**	**0.92**	0.908	0.582	0.741	0.812	0.88	0.881	**0.918**	**0.909**	**0.906**	0.406	0.545	0.616	0.706	0.693	**0.754**	**0.731**	**0.734**
	wikiSERA-10	**0.806**	**0.834**	**0.889**	0.91	**0.9**	0.906	0.895	0.896	**0.769**	**0.81**	**0.852**	0.897	0.88	0.884	0.877	0.896	**0.568**	**0.623**	**0.671**	0.73	0.695	0.705	0.693	0.713
	wikiSERA-DIS-5	0.669	0.761	0.827	0.897	0.871	0.901	0.916	0.908	0.608	0.745	0.802	0.885	0.861	0.9	0.896	0.893	0.425	0.55	0.601	0.712	0.665	0.719	0.723	0.724
	wikiSERA-DIS-10	0.772	0.818	0.873	**0.913**	0.894	0.904	0.913	**0.909**	0.746	0.804	0.835	**0.908**	**0.886**	0.898	0.89	0.899	0.543	0.609	0.661	**0.745**	**0.71**	0.718	0.714	0.718

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.652	0.768	0.812	**0.912**	0.869	**0.912**	**0.906**	0.874	0.552	0.732	0.811	0.894	0.871	0.882	**0.904**	0.873	0.388	0.537	0.618	**0.724**	0.684	0.704	**0.726**	0.696
	SERA-10	0.827	0.831	0.9	0.905	0.898	0.905	0.88	0.869	0.792	0.807	0.879	0.887	0.878	0.882	0.864	0.869	**0.592**	0.606	0.707	0.722	0.685	0.697	0.687	0.671
	SERA-NP-5	0.689	0.79	0.814	0.869	0.871	0.893	0.863	0.905	0.632	0.755	0.805	0.864	0.868	0.867	0.855	**0.911**	0.43	0.561	0.6	0.687	0.681	0.695	0.67	**0.736**
	SERA-NP-10	**0.828**	**0.836**	0.893	0.905	**0.914**	0.892	0.902	0.886	0.788	0.813	**0.882**	0.891	**0.916**	0.87	0.898	0.88	0.58	**0.627**	0.701	0.716	**0.756**	0.69	0.724	0.701
	SERA-KW-5	0.645	0.767	0.807	0.911	0.868	0.91	0.9	0.894	0.551	0.737	0.8	0.887	0.867	0.878	0.887	0.859	0.388	0.539	0.606	0.721	0.685	0.692	0.707	0.685
	SERA-KW-10	0.824	0.829	**0.901**	0.905	0.894	0.903	0.883	**0.912**	**0.798**	0.813	0.877	0.886	0.876	0.883	0.873	0.893	0.591	0.609	**0.709**	0.716	0.684	0.696	0.695	0.725
	SERA-DIS-5	0.626	0.75	0.821	0.909	0.852	0.891	0.899	0.892	0.53	0.726	0.813	0.883	0.851	0.864	0.885	0.886	0.366	0.53	0.615	0.716	0.654	0.674	0.71	0.722
	SERA-DIS-10	0.779	0.814	0.886	0.9	0.892	0.905	0.895	0.891	0.745	0.815	0.866	0.881	0.885	**0.892**	0.883	0.889	0.539	0.609	0.682	0.717	0.698	**0.714**	0.704	0.713
	SERA-DIS-NP-5	0.678	0.774	0.79	0.871	0.877	0.871	0.864	0.909	0.624	0.746	0.797	0.856	0.858	0.851	0.861	0.893	0.429	0.549	0.588	0.689	0.673	0.672	0.676	0.711
	SERA-DIS-NP-10	0.775	0.811	0.866	0.903	0.908	0.892	0.896	0.899	0.748	0.788	0.861	**0.896**	0.9	0.881	0.888	0.879	0.539	0.59	0.676	**0.724**	0.728	0.708	0.721	0.701
	SERA-DIS-KW-5	0.611	0.754	0.817	0.904	0.849	0.888	0.895	0.896	0.505	0.738	0.81	0.88	0.852	0.869	0.876	0.862	0.354	0.543	0.607	0.71	0.658	0.675	0.699	0.691
	SERA-DIS-KW-10	0.773	0.818	0.886	0.894	0.89	0.901	0.896	**0.912**	0.734	**0.816**	0.868	0.878	0.882	0.89	0.884	0.886	0.531	0.609	0.677	0.706	0.693	0.708	0.707	0.718
	wikiSERA-5	0.667	0.765	0.836	0.904	0.886	**0.917**	**0.92**	**0.908**	0.581	0.741	0.812	0.878	0.877	**0.917**	**0.908**	**0.904**	0.406	0.545	0.616	0.706	0.688	**0.752**	**0.731**	**0.731**
	wikiSERA-10	**0.806**	**0.834**	**0.889**	0.909	**0.899**	0.906	0.894	0.896	**0.765**	**0.807**	**0.85**	0.894	0.878	0.881	0.875	0.893	**0.565**	**0.623**	**0.671**	0.73	0.693	0.702	0.693	0.711
	wikiSERA-DIS-5	0.67	0.761	0.827	0.897	0.87	0.901	0.916	0.907	0.606	0.744	0.802	0.883	0.857	0.898	0.894	0.891	0.425	0.547	0.601	0.712	0.66	0.719	0.723	0.722
	wikiSERA-DIS-10	0.772	0.817	0.873	**0.913**	0.893	0.904	0.913	**0.908**	0.744	0.801	0.834	**0.905**	**0.884**	0.896	0.888	0.895	0.54	0.607	0.659	**0.742**	**0.707**	0.716	0.714	0.716

Table A.9: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.649	0.763	0.823	0.908	0.88	**0.912**	**0.913**	0.876	0.549	0.727	0.818	0.89	0.881	0.886	**0.908**	0.874	0.383	0.532	0.627	0.726	0.695	0.708	0.732	0.698
	SERA-10	**0.823**	0.835	0.899	0.907	0.902	0.907	0.887	0.878	**0.791**	0.822	0.882	0.893	0.891	0.887	0.871	0.88	**0.594**	0.62	0.706	0.723	0.706	0.704	0.693	0.696
	SERA-NP-5	0.689	0.802	0.816	0.881	0.888	0.894	0.866	0.895	0.63	0.801	0.828	0.875	0.879	0.871	0.863	0.896	0.437	0.605	0.633	0.695	0.693	0.706	0.681	0.717
	SERA-NP-10	0.818	**0.847**	0.9	0.905	**0.917**	0.896	0.908	0.886	0.778	0.828	**0.897**	0.891	**0.912**	0.879	0.905	0.884	0.566	**0.628**	**0.721**	0.711	**0.745**	0.704	**0.739**	0.704
	SERA-KW-5	0.647	0.763	0.824	**0.91**	0.877	0.909	0.909	0.894	0.539	0.735	0.823	0.889	0.878	0.883	0.901	0.86	0.379	0.542	0.63	0.723	0.702	0.704	0.721	0.692
	SERA-KW-10	0.818	0.835	**0.901**	0.906	0.899	0.905	0.89	0.908	**0.791**	0.823	0.884	0.893	0.885	0.888	0.88	**0.898**	0.587	0.62	0.711	0.724	0.701	0.704	0.705	**0.735**
	SERA-DIS-5	0.636	0.753	0.821	0.904	0.866	0.892	0.905	0.895	0.532	0.728	0.818	0.888	0.87	0.866	0.885	0.888	0.374	0.532	0.624	0.724	0.676	0.683	0.713	0.712
	SERA-DIS-10	0.78	0.823	0.887	0.9	0.897	0.905	0.9	0.899	0.733	0.819	0.871	0.89	0.897	**0.896**	0.888	**0.898**	0.538	0.614	0.684	**0.728**	0.711	0.716	0.711	0.722
	SERA-DIS-NP-5	0.676	0.785	0.798	0.885	0.89	0.88	0.875	0.9	0.62	0.785	0.817	0.868	0.882	0.859	0.864	0.878	0.428	0.578	0.61	0.69	0.701	0.685	0.679	0.706
	SERA-DIS-NP-10	0.767	0.828	0.874	0.908	0.915	0.897	0.907	0.895	0.738	0.823	0.881	**0.901**	0.897	0.889	0.905	0.879	0.534	0.62	0.702	0.727	0.721	**0.718**	0.735	0.695
	SERA-DIS-KW-5	0.626	0.757	0.822	0.902	0.862	0.889	0.903	0.9	0.527	0.738	0.813	0.885	0.862	0.871	0.885	0.874	0.37	0.543	0.614	0.721	0.664	0.682	0.712	0.704
	SERA-DIS-KW-10	0.775	0.829	0.888	0.895	0.893	0.902	0.902	**0.912**	0.731	**0.831**	0.872	0.884	0.893	0.892	0.888	0.895	0.532	0.627	0.682	0.718	0.704	0.711	0.709	0.73
	wikiSERA-5	0.668	0.76	0.849	0.903	0.893	**0.916**	**0.928**	0.91	0.587	0.732	0.827	0.885	**0.893**	**0.912**	**0.924**	0.903	0.408	0.537	0.635	0.711	**0.719**	**0.741**	**0.76**	**0.73**
	wikiSERA-10	**0.812**	**0.842**	**0.895**	0.911	**0.906**	0.91	0.902	0.898	**0.783**	**0.832**	**0.872**	0.892	0.885	0.897	0.89	0.897	**0.578**	**0.635**	**0.695**	0.723	0.707	0.717	0.708	0.714
	wikiSERA-DIS-5	0.675	0.751	0.841	0.901	0.885	0.907	0.924	**0.913**	0.618	0.738	0.825	0.885	0.885	0.904	0.91	0.893	0.438	0.54	0.631	0.709	0.7	0.734	0.746	0.721
	wikiSERA-DIS-10	0.78	0.828	0.887	**0.915**	0.903	0.91	0.918	0.912	0.753	0.826	0.854	**0.909**	**0.893**	0.899	0.896	**0.904**	0.549	0.623	0.673	**0.75**	0.718	0.727	0.724	0.722

	Method	Pearson								Spearman								Kendall							
		TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.65	0.763	0.823	0.907	0.878	**0.911**	**0.912**	0.876	0.549	0.727	0.819	0.889	0.877	0.883	**0.906**	0.873	0.386	0.532	0.627	0.726	0.69	0.705	0.732	0.698
	SERA-10	**0.824**	0.835	0.899	0.906	0.902	0.906	0.887	0.878	**0.789**	0.82	0.88	0.89	0.888	0.883	0.868	0.878	**0.591**	0.618	0.706	0.721	0.703	0.702	0.693	0.694
	SERA-NP-5	0.69	0.802	0.816	0.881	0.887	0.893	0.866	0.894	0.63	0.8	0.826	0.874	0.876	0.868	0.86	**0.895**	0.434	0.602	0.63	0.695	0.691	0.704	0.678	0.717
	SERA-NP-10	0.819	**0.847**	**0.9**	0.905	**0.916**	0.895	0.908	0.886	0.777	0.827	**0.895**	0.887	**0.91**	0.875	0.903	0.882	0.566	**0.628**	**0.721**	0.709	**0.742**	0.702	**0.737**	0.704
	SERA-KW-5	0.648	0.763	0.824	**0.909**	0.876	0.908	0.908	0.893	0.539	0.735	0.824	0.887	0.874	0.88	0.898	0.857	0.381	0.542	0.63	0.723	0.697	0.702	0.721	0.692
	SERA-KW-10	0.819	0.835	**0.9**	0.906	0.898	0.904	0.89	0.907	0.788	0.821	0.882	0.89	0.882	0.884	0.878	**0.895**	0.584	0.617	0.708	0.722	0.699	0.702	0.702	**0.733**
	SERA-DIS-5	0.637	0.753	0.821	0.904	0.865	0.891	0.905	0.895	0.532	0.727	0.817	0.887	0.867	0.863	0.882	0.886	0.374	0.532	0.624	0.724	0.673	0.681	0.711	0.712
	SERA-DIS-10	0.78	0.823	0.887	0.899	0.896	0.904	0.9	0.899	0.731	0.817	0.87	0.887	0.895	**0.893**	0.885	**0.895**	0.535	0.612	0.684	**0.728**	0.708	0.714	0.708	0.719
	SERA-DIS-NP-5	0.677	0.785	0.797	0.885	0.889	0.88	0.875	0.9	0.621	0.783	0.815	0.867	0.878	0.856	0.861	0.876	0.428	0.575	0.608	0.69	0.696	0.683	0.677	0.704
	SERA-DIS-NP-10	0.768	0.828	0.874	0.908	0.914	0.896	0.907	0.894	0.738	0.822	0.879	**0.898**	0.895	0.886	0.903	0.876	0.534	0.62	0.702	0.724	0.718	**0.715**	0.733	0.693
	SERA-DIS-KW-5	0.627	0.758	0.822	0.902	0.861	0.888	0.902	0.899	0.527	0.738	0.813	0.884	0.859	0.868	0.882	0.871	0.37	0.543	0.612	0.721	0.661	0.679	0.71	0.704
	SERA-DIS-KW-10	0.775	0.829	0.888	0.895	0.893	0.901	0.901	**0.911**	0.73	**0.829**	0.87	0.881	0.89	0.889	0.885	0.892	0.532	0.625	0.682	0.718	0.701	0.708	0.707	0.728
	wikiSERA-5	0.669	0.761	0.849	0.903	0.892	**0.915**	**0.928**	0.909	0.587	0.732	0.827	0.883	0.889	**0.91**	**0.923**	**0.901**	0.408	0.536	0.635	0.711	0.714	**0.739**	**0.76**	**0.728**
	wikiSERA-10	**0.812**	**0.842**	**0.895**	0.91	**0.905**	0.909	0.901	0.897	**0.779**	**0.83**	**0.87**	0.89	0.883	0.894	0.888	0.894	**0.576**	**0.633**	**0.695**	0.723	0.704	0.715	0.708	0.711
	wikiSERA-DIS-5	0.676	0.751	0.841	0.901	0.884	0.906	0.924	**0.912**	0.617	0.737	0.825	0.883	0.881	0.902	0.908	0.891	0.438	0.538	0.629	0.709	0.695	0.731	0.743	0.718
	wikiSERA-DIS-10	0.78	0.828	0.886	**0.914**	0.902	0.909	0.918	**0.912**	0.751	0.824	0.852	**0.906**	**0.89**	0.897	0.894	**0.901**	0.546	0.62	0.671	**0.75**	**0.716**	0.724	0.722	0.719

Table A.10: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.2 Correlation of SERA and wikiSERA with Responsiveness on TAC2008/AQUAINT-2

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.528	0.598	0.698	0.737	0.783	0.822	**0.838**	0.795	0.361	0.508	0.654	0.713	0.726	**0.804**	0.797	0.775	0.258	0.369	0.472	0.533	0.54	0.609	0.607	0.584
SERA-10	**0.681**	0.698	0.748	**0.801**	0.81	**0.831**	0.824	0.807	**0.585**	0.665	0.756	**0.803**	**0.786**	0.777	0.769	0.767	**0.426**	0.488	0.558	**0.604**	**0.598**	0.579	0.582	0.578
SERA-NP-5	0.558	0.674	0.631	0.771	0.794	0.788	0.747	0.757	0.429	0.579	0.615	0.739	0.753	0.752	0.732	0.735	0.303	0.414	0.437	0.548	0.568	0.559	0.548	0.545
SERA-NP-10	0.655	**0.735**	**0.784**	0.79	0.811	0.818	0.831	0.771	0.561	**0.693**	**0.79**	0.746	0.747	0.796	0.812	0.731	0.409	**0.504**	**0.581**	0.552	0.564	0.596	0.614	0.536
SERA-KW-5	0.534	0.599	0.719	0.752	0.783	0.813	0.836	0.797	0.378	0.517	0.695	0.713	0.727	0.797	0.804	0.725	0.268	0.376	0.505	0.526	0.535	0.607	0.613	0.536
SERA-KW-10	0.668	0.707	0.752	0.79	**0.813**	0.828	0.828	0.789	0.55	0.675	0.743	0.786	0.779	0.774	0.776	0.774	0.396	0.494	0.546	0.587	0.588	0.568	0.589	0.588
SERA-DIS-5	0.544	0.61	0.655	0.71	0.763	0.814	0.821	0.795	0.346	0.529	0.632	0.661	0.721	0.798	0.768	0.759	0.24	0.383	0.45	0.462	0.534	**0.611**	0.581	0.561
SERA-DIS-10	0.664	0.699	0.732	0.771	0.789	0.821	0.826	**0.818**	0.528	0.645	0.737	0.748	0.775	0.761	0.779	**0.796**	0.374	0.462	0.547	0.546	0.581	0.568	0.588	**0.594**
SERA-DIS-NP-5	0.532	0.664	0.647	0.77	0.762	0.786	0.76	0.706	0.409	0.582	0.649	0.742	0.715	0.756	0.749	0.666	0.269	0.411	0.448	0.533	0.529	0.554	0.554	0.497
SERA-DIS-NP-10	0.608	0.732	0.764	0.788	**0.813**	0.814	0.837	0.749	0.503	0.683	0.778	0.76	0.777	0.782	**0.828**	0.721	0.365	0.48	0.563	0.554	0.586	0.585	**0.636**	0.528
SERA-DIS-KW-5	0.543	0.615	0.675	0.724	0.761	0.8	0.816	0.799	0.369	0.53	0.647	0.684	0.693	0.792	0.77	0.743	0.259	0.385	0.467	0.493	0.508	0.601	0.585	0.54
SERA-DIS-KW-10	0.656	0.713	0.742	0.767	0.795	0.818	0.831	0.807	0.514	0.678	0.736	0.728	0.784	0.763	0.776	0.775	0.357	0.487	0.538	0.523	0.594	0.568	0.596	0.587
wikiSERA-5	0.553	0.587	0.726	0.747	0.766	0.798	**0.866**	**0.83**	0.422	0.475	0.688	0.732	0.75	0.764	**0.829**	**0.797**	0.292	0.326	0.496	0.529	0.553	0.576	**0.647**	**0.611**
wikiSERA-10	**0.707**	**0.72**	0.76	**0.796**	0.814	**0.835**	0.837	0.8	**0.645**	**0.683**	**0.756**	**0.782**	0.78	0.778	0.774	0.779	**0.462**	**0.494**	**0.56**	**0.591**	0.586	0.576	0.591	0.58
wikiSERA-DIS-5	0.565	0.548	0.721	0.735	0.77	0.801	0.844	0.821	0.445	0.448	0.677	0.71	0.747	**0.794**	0.79	0.765	0.315	0.312	0.488	0.502	0.554	**0.605**	0.602	0.565
wikiSERA-DIS-10	0.685	0.712	**0.761**	0.777	**0.815**	0.825	0.837	0.817	0.583	0.667	0.745	0.764	**0.791**	0.773	0.761	0.784	0.421	0.474	0.537	0.558	**0.603**	0.573	0.573	0.585

Table A.11: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.55	0.61	0.658	0.773	0.774	0.817	0.82	0.787	0.403	0.531	0.602	0.722	0.744	0.769	**0.808**	0.785	0.295	0.378	0.427	0.53	0.549	0.569	**0.622**	0.597
SERA-10	**0.722**	**0.69**	0.774	0.826	0.766	0.831	0.787	0.779	0.683	**0.651**	0.735	0.802	0.729	0.801	0.738	0.739	0.483	**0.474**	0.558	0.609	0.539	0.603	0.537	0.543
SERA-NP-5	0.551	0.668	0.688	0.732	0.753	0.752	0.746	0.841	0.434	0.63	0.655	0.687	0.697	0.7	0.702	**0.846**	0.321	0.451	0.468	0.495	0.503	0.505	0.506	**0.666**
SERA-NP-10	0.697	**0.69**	**0.777**	**0.829**	**0.808**	0.813	**0.825**	0.794	**0.687**	0.604	0.738	0.792	**0.771**	0.773	0.791	0.764	**0.49**	0.439	0.548	0.608	**0.585**	0.574	0.595	0.565
SERA-KW-5	0.543	0.611	0.641	0.776	0.769	0.798	0.816	0.829	0.393	0.534	0.581	0.725	0.739	0.759	0.787	0.773	0.283	0.382	0.407	0.536	0.548	0.556	0.606	0.584
SERA-KW-10	**0.722**	0.685	**0.777**	0.827	0.765	**0.832**	0.796	**0.856**	0.686	0.627	0.742	**0.804**	0.729	**0.808**	0.75	0.805	**0.49**	0.451	0.569	**0.61**	0.54	**0.607**	0.558	0.61
SERA-DIS-5	0.507	0.587	0.67	0.776	0.758	0.802	0.806	0.776	0.367	0.497	0.624	0.705	0.707	0.786	0.786	0.765	0.258	0.363	0.446	0.512	0.512	0.588	0.591	0.57
SERA-DIS-10	0.668	0.645	0.775	0.803	0.754	0.819	0.798	0.771	0.614	0.599	**0.744**	0.766	0.726	0.805	0.765	0.737	0.436	0.436	**0.571**	0.563	0.53	0.605	0.563	0.533
SERA-DIS-NP-5	0.546	0.658	0.677	0.733	0.731	0.712	0.751	0.816	0.43	0.614	0.646	0.67	0.652	0.658	0.733	0.791	0.308	0.449	0.478	0.482	0.471	0.462	0.542	0.624
SERA-DIS-NP-10	0.628	0.677	0.754	0.817	0.786	0.794	0.806	0.787	0.553	0.599	0.706	0.777	0.723	0.747	0.769	0.736	0.392	0.428	0.51	0.588	0.531	0.554	0.573	0.547
SERA-DIS-KW-5	0.489	0.59	0.645	0.764	0.75	0.787	0.809	0.804	0.335	0.507	0.597	0.709	0.703	0.777	0.774	0.754	0.241	0.365	0.417	0.51	0.506	0.574	0.575	0.563
SERA-DIS-KW-10	0.668	0.651	0.772	0.797	0.754	0.812	0.809	0.827	0.623	0.605	0.743	0.76	0.724	0.807	0.774	0.766	0.444	0.439	0.562	0.553	0.527	0.605	0.564	0.574
wikiSERA-5	0.528	0.608	0.722	0.827	0.761	0.827	**0.836**	**0.827**	0.411	0.547	0.648	0.794	0.723	0.788	**0.809**	**0.824**	0.287	0.395	0.463	0.606	0.524	0.601	**0.625**	**0.629**
wikiSERA-10	**0.69**	**0.702**	**0.766**	0.844	**0.78**	**0.832**	0.797	0.815	**0.638**	**0.665**	**0.705**	**0.816**	**0.749**	0.795	0.737	0.791	**0.438**	**0.487**	**0.535**	**0.631**	**0.554**	0.603	0.533	0.59
wikiSERA-DIS-5	0.541	0.601	0.711	0.817	0.754	0.805	0.825	0.818	0.46	0.549	0.624	0.76	0.709	**0.801**	0.797	0.802	0.317	0.393	0.448	0.567	0.524	**0.604**	0.598	0.615
wikiSERA-DIS-10	0.634	0.666	0.759	**0.847**	0.765	0.803	0.81	0.808	0.567	0.634	0.698	0.811	0.733	0.785	0.762	0.79	0.387	0.457	0.528	0.613	0.54	0.582	0.567	0.597

Table A.12: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.49	0.597	0.655	0.792	0.707	**0.833**	**0.828**	0.814	0.318	0.549	0.634	0.758	0.648	0.765	0.785	0.784	0.219	0.395	0.438	0.574	0.461	0.574	0.593	0.59
SERA-10	**0.688**	**0.69**	0.779	**0.816**	0.816	0.827	0.808	0.782	**0.621**	**0.648**	**0.757**	**0.797**	0.787	0.78	0.761	0.752	**0.447**	**0.457**	0.563	**0.594**	0.595	0.58	0.567	0.553
SERA-NP-5	0.545	0.606	0.677	0.725	0.76	0.818	0.742	0.795	0.442	0.598	0.671	0.725	0.723	**0.797**	0.718	0.799	0.3	0.436	0.463	0.534	0.534	**0.591**	0.518	**0.608**
SERA-NP-10	0.675	0.665	0.771	0.814	**0.842**	0.81	0.818	0.792	0.596	0.632	0.753	0.788	**0.811**	0.788	**0.81**	0.766	0.43	0.452	0.558	0.588	**0.615**	0.583	**0.604**	0.564
SERA-KW-5	0.473	0.593	0.654	0.797	0.711	0.825	0.822	0.816	0.301	0.55	0.636	0.759	0.658	0.758	0.771	0.743	0.211	0.395	0.438	0.574	0.463	0.56	0.576	0.559
SERA-KW-10	0.678	0.682	**0.781**	0.812	0.808	0.815	0.81	0.822	0.605	0.634	0.754	0.785	0.784	0.766	0.774	0.782	0.443	0.442	**0.564**	0.578	0.59	0.563	0.581	0.588
SERA-DIS-5	0.485	0.572	0.677	0.795	0.668	0.781	0.816	**0.826**	0.318	0.547	0.641	0.769	0.608	0.733	0.769	**0.803**	0.217	0.382	0.451	0.576	0.426	0.54	0.565	0.605
SERA-DIS-10	0.61	0.669	0.764	0.804	0.801	0.813	0.818	0.813	0.538	0.637	0.735	0.783	0.774	0.768	0.779	0.773	0.376	0.437	0.539	0.585	0.568	0.57	0.577	0.57
SERA-DIS-NP-5	0.527	0.561	0.64	0.732	0.742	0.804	0.743	0.768	0.437	0.55	0.652	0.731	0.7	0.785	0.737	0.751	0.298	0.396	0.448	0.544	0.489	0.58	0.529	0.539
SERA-DIS-NP-10	0.643	0.61	0.734	0.79	0.832	0.816	0.817	0.793	0.604	0.603	0.709	0.779	0.793	0.795	0.802	0.758	0.427	0.423	0.501	0.581	0.592	0.588	0.601	0.569
SERA-DIS-KW-5	0.483	0.572	0.677	0.795	0.673	0.768	0.808	0.811	0.347	0.559	0.647	0.762	0.621	0.715	0.755	0.762	0.243	0.396	0.448	0.564	0.429	0.525	0.547	0.574
SERA-DIS-KW-10	0.603	0.666	0.76	0.795	0.799	0.802	0.811	0.815	0.521	0.633	0.74	0.782	0.778	0.744	0.774	0.769	0.356	0.437	0.541	0.576	0.569	0.547	0.57	0.581
wikiSERA-5	0.496	0.566	0.663	0.753	0.761	0.819	**0.85**	**0.845**	0.366	0.525	0.626	0.712	0.719	0.792	0.801	**0.832**	0.251	0.369	0.45	0.513	0.525	0.593	0.61	**0.635**
wikiSERA-10	**0.685**	**0.644**	**0.781**	**0.826**	**0.811**	**0.829**	0.82	0.815	**0.605**	**0.59**	**0.739**	0.809	**0.775**	0.778	0.782	0.803	**0.427**	**0.416**	**0.546**	0.611	**0.584**	0.581	0.586	0.607
wikiSERA-DIS-5	0.524	0.543	0.644	0.762	0.706	0.801	0.843	0.828	0.445	0.523	0.601	0.755	0.655	0.781	0.804	0.786	0.308	0.357	0.428	0.547	0.462	0.586	0.601	0.598
wikiSERA-DIS-10	0.641	0.608	0.748	0.812	0.785	**0.829**	0.839	0.84	0.589	0.561	0.69	**0.816**	0.764	**0.8**	**0.81**	0.804	0.423	0.391	0.505	**0.614**	0.559	**0.608**	**0.621**	0.613

Table A.13: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_3$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.501	0.645	0.71	**0.826**	0.793	0.82	0.802	0.783	0.323	0.554	0.675	**0.764**	0.762	0.767	0.774	0.778	0.226	0.405	0.489	**0.579**	0.574	0.57	0.575	0.571
SERA-10	0.7	0.704	0.813	0.78	**0.835**	**0.837**	0.797	0.808	0.612	0.64	**0.776**	0.75	0.795	0.783	0.75	0.791	0.437	0.47	**0.572**	0.558	**0.608**	0.578	0.545	0.588
SERA-NP-5	0.533	0.617	0.65	0.779	0.769	0.79	0.803	0.824	0.404	0.522	0.643	0.732	0.691	0.716	0.732	0.795	0.283	0.376	0.447	0.546	0.512	0.529	0.544	0.595
SERA-NP-10	**0.709**	**0.739**	0.775	0.794	0.809	0.817	**0.854**	0.811	**0.653**	**0.683**	0.73	**0.764**	0.786	0.752	**0.8**	**0.796**	**0.471**	**0.497**	0.538	0.564	0.587	0.548	**0.604**	0.599
SERA-KW-5	0.496	0.647	0.717	0.82	0.797	0.826	0.795	0.777	0.331	0.558	0.673	0.752	0.764	0.775	0.766	0.728	0.235	0.413	0.494	0.572	0.571	0.578	0.568	0.533
SERA-KW-10	0.697	0.708	**0.817**	0.777	0.83	0.831	0.804	**0.842**	0.613	0.637	0.773	0.738	0.794	0.776	0.765	0.789	0.441	0.469	0.57	0.545	0.606	0.573	0.559	**0.609**
SERA-DIS-5	0.448	0.618	0.684	0.799	0.799	0.784	0.807	0.798	0.235	0.546	0.659	0.759	0.775	0.727	0.774	0.778	0.181	0.399	0.478	0.567	0.58	0.529	0.564	0.564
SERA-DIS-10	0.634	0.699	0.777	0.773	0.834	0.825	0.811	0.809	0.538	0.643	0.743	0.741	**0.797**	0.78	0.793	0.792	0.386	0.466	0.545	0.57	0.605	0.577	0.587	0.586
SERA-DIS-NP-5	0.529	0.583	0.612	0.768	0.81	0.75	0.79	0.822	0.384	0.484	0.603	0.732	0.763	0.685	0.736	0.757	0.258	0.36	0.421	0.546	0.573	0.484	0.534	0.562
SERA-DIS-NP-10	0.62	0.691	0.729	0.775	0.813	0.806	0.823	0.837	0.55	0.602	0.714	0.741	0.781	0.775	0.772	0.784	0.383	0.431	0.525	0.537	0.588	0.564	0.57	0.592
SERA-DIS-KW-5	0.415	0.628	0.702	0.794	0.798	0.796	0.802	0.795	0.2	0.562	0.668	0.753	0.769	0.748	0.75	0.742	0.157	0.414	0.48	0.565	0.565	0.548	0.539	0.539
SERA-DIS-KW-10	0.624	0.709	0.791	0.763	0.833	0.83	0.812	0.833	0.532	0.641	0.761	0.733	**0.797**	**0.793**	0.788	0.768	0.38	0.462	0.561	0.556	0.598	**0.587**	0.585	0.577
wikiSERA-5	0.565	0.663	0.731	**0.809**	0.819	0.796	0.831	0.787	0.399	0.617	0.696	0.724	0.765	**0.801**	0.812	0.753	0.281	0.452	0.514	**0.548**	0.577	**0.608**	0.614	0.559
wikiSERA-10	**0.672**	**0.723**	**0.803**	0.78	**0.83**	**0.831**	0.852	**0.819**	**0.597**	0.667	**0.75**	**0.74**	**0.788**	0.776	**0.828**	**0.777**	**0.418**	**0.49**	**0.557**	0.545	**0.608**	0.571	**0.639**	**0.574**
wikiSERA-DIS-5	0.545	0.668	0.705	0.766	0.801	0.765	0.828	0.767	0.363	0.647	0.673	0.696	0.749	0.762	0.773	0.7	0.25	0.464	0.49	0.516	0.557	0.553	0.565	0.502
wikiSERA-DIS-10	0.646	0.717	0.77	0.776	0.82	0.816	**0.854**	0.808	0.55	**0.668**	0.721	0.732	0.763	0.796	0.814	0.773	0.389	0.483	0.527	0.537	0.582	0.585	0.611	0.57

Table A.14: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_4$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.537	0.631	0.692	0.797	0.781	**0.846**	**0.848**	0.82	0.383	0.555	0.674	0.78	0.744	**0.809**	0.813	0.805	0.272	0.393	0.488	0.59	0.537	0.609	0.617	0.611
SERA-10	**0.713**	0.713	0.781	0.826	0.813	0.843	0.82	0.807	**0.641**	0.687	0.756	0.805	0.794	0.791	0.771	0.768	**0.464**	0.497	0.558	0.608	0.602	0.579	0.579	0.568
SERA-NP-5	0.567	0.689	0.694	0.77	0.795	0.817	0.772	0.825	0.467	0.659	0.709	0.77	0.757	0.772	0.758	**0.821**	0.335	0.485	0.507	0.58	0.561	0.576	0.561	**0.629**
SERA-NP-10	0.691	**0.723**	**0.796**	**0.83**	**0.838**	0.826	0.842	0.803	0.624	0.69	**0.784**	**0.815**	**0.811**	0.787	0.825	0.767	0.448	**0.509**	**0.59**	**0.616**	**0.621**	0.581	0.63	0.567
SERA-KW-5	0.532	0.629	0.693	0.805	0.778	0.836	0.846	0.832	0.386	0.558	0.679	0.797	0.742	0.799	0.812	0.78	0.272	0.407	0.491	0.607	0.541	0.602	0.621	0.587
SERA-KW-10	0.706	0.711	0.784	0.822	0.811	0.839	0.826	**0.833**	0.63	**0.697**	0.756	0.8	0.792	0.792	0.776	0.802	0.45	0.507	0.559	0.603	0.602	0.577	0.582	0.61
SERA-DIS-5	0.528	0.624	0.7	0.799	0.762	0.822	0.839	0.826	0.365	0.557	0.673	0.776	0.729	0.803	0.798	0.805	0.256	0.404	0.479	0.58	0.524	**0.613**	0.596	0.601
SERA-DIS-10	0.664	0.694	0.777	0.812	0.799	0.835	0.831	0.821	0.596	0.669	0.755	0.799	0.782	0.802	0.798	0.792	0.426	0.483	0.563	0.61	0.581	0.605	0.597	0.593
SERA-DIS-NP-5	0.553	0.668	0.692	0.782	0.778	0.804	0.788	0.809	0.434	0.653	0.72	0.77	0.729	0.768	0.765	0.78	0.301	0.477	0.517	0.579	0.534	0.569	0.573	0.584
SERA-DIS-NP-10	0.642	0.702	0.773	0.825	0.832	0.824	0.842	0.8	0.567	0.682	0.759	0.812	0.787	0.784	**0.834**	0.767	0.398	0.492	0.561	0.611	0.59	0.581	**0.636**	0.571
SERA-DIS-KW-5	0.521	0.627	0.698	0.8	0.76	0.81	0.836	0.827	0.379	0.574	0.684	0.771	0.722	0.791	0.797	0.772	0.269	0.414	0.484	0.575	0.519	0.597	0.597	0.579
SERA-DIS-KW-10	0.661	0.7	0.777	0.806	0.8	0.828	0.834	0.83	0.594	0.684	0.756	0.791	0.776	0.796	0.801	0.786	0.422	0.49	0.562	0.593	0.575	0.597	0.603	0.596
wikiSERA-5	0.543	0.619	0.729	0.807	0.784	0.842	**0.872**	**0.855**	0.443	0.565	0.683	0.79	0.756	0.818	**0.851**	**0.846**	0.308	0.411	0.503	0.597	0.559	0.628	**0.658**	**0.654**
wikiSERA-10	**0.71**	**0.71**	**0.787**	**0.838**	**0.816**	**0.847**	0.833	0.825	**0.658**	**0.693**	**0.752**	0.826	0.783	0.81	0.786	0.81	**0.463**	**0.499**	0.558	0.628	**0.601**	0.607	0.597	0.611
wikiSERA-DIS-5	0.56	0.603	0.726	0.813	0.775	0.831	0.863	0.85	0.472	0.562	0.686	0.8	0.746	**0.825**	0.842	0.827	0.338	0.408	0.504	0.61	0.552	**0.637**	0.65	0.635
wikiSERA-DIS-10	0.673	0.687	0.782	0.837	0.805	0.837	0.846	0.84	0.614	0.664	0.745	**0.828**	**0.784**	0.81	0.791	0.815	0.439	0.48	**0.561**	**0.631**	0.59	0.608	0.601	0.622

Table A.15: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.539	0.646	0.711	0.808	0.807	0.841	0.842	0.811	0.367	0.56	0.693	0.767	0.777	0.803	0.823	0.799	0.271	0.402	0.511	0.571	0.576	**0.603**	0.616	0.601
SERA-10	**0.719**	0.713	0.793	0.814	0.819	**0.845**	0.816	0.815	0.657	0.675	0.777	**0.794**	0.779	0.794	0.766	0.775	0.47	0.491	**0.582**	**0.599**	0.593	0.59	0.567	0.571
SERA-NP-5	0.566	0.697	0.688	0.788	0.797	0.8	0.796	0.832	0.438	0.635	0.685	0.773	0.739	0.745	0.755	**0.813**	0.314	0.462	0.489	0.583	0.55	0.552	0.556	**0.625**
SERA-NP-10	0.704	**0.743**	**0.799**	**0.821**	**0.827**	0.826	**0.852**	0.809	0.648	**0.688**	**0.783**	0.782	**0.801**	0.77	**0.829**	0.787	0.467	**0.5**	**0.582**	0.586	**0.611**	0.568	**0.634**	0.587
SERA-KW-5	0.538	0.647	0.715	0.813	0.803	0.836	0.838	0.816	0.355	0.582	0.694	0.781	0.775	0.794	0.809	0.762	0.257	0.429	0.508	0.588	0.583	0.588	0.615	0.566
SERA-KW-10	0.714	0.716	0.797	0.811	0.817	0.842	0.824	**0.842**	**0.663**	0.664	0.777	0.786	0.776	0.793	0.776	0.796	**0.478**	0.479	0.58	0.59	0.592	0.584	0.578	0.61
SERA-DIS-5	0.515	0.639	0.699	0.799	0.803	0.824	0.839	0.816	0.335	0.571	0.69	0.76	0.771	0.796	0.81	0.796	0.239	0.408	0.499	0.57	0.577	0.6	0.605	0.588
SERA-DIS-10	0.672	0.701	0.78	0.801	0.81	0.837	0.828	0.82	0.604	0.666	0.764	0.777	0.785	0.796	0.793	0.794	0.427	0.484	0.567	0.588	0.586	0.59	0.588	0.588
SERA-DIS-NP-5	0.554	0.678	0.681	0.794	0.798	0.779	0.807	0.822	0.419	0.613	0.689	0.761	0.75	0.719	0.779	0.771	0.298	0.439	0.499	0.575	0.559	0.529	0.579	0.581
SERA-DIS-NP-10	0.636	0.723	0.772	0.82	0.825	0.819	0.844	0.817	0.559	0.661	0.759	0.787	0.791	0.767	0.812	0.783	0.393	0.471	0.557	0.589	0.597	0.564	0.616	0.588
SERA-DIS-KW-5	0.497	0.645	0.704	0.797	0.798	0.821	0.838	0.822	0.298	0.579	0.693	0.769	0.763	**0.804**	0.8	0.77	0.216	0.42	0.502	0.577	0.568	0.599	0.597	0.573
SERA-DIS-KW-10	0.668	0.711	0.788	0.795	0.809	0.835	0.834	0.836	0.596	0.675	0.769	0.771	0.782	0.8	0.797	0.789	0.423	0.487	0.574	0.577	0.592	0.596	0.602	0.597
wikiSERA-5	0.567	0.654	0.755	0.821	0.809	0.835	**0.867**	**0.838**	0.429	0.6	0.713	0.79	0.773	0.817	**0.853**	**0.826**	0.307	0.431	0.527	0.599	0.574	0.621	**0.664**	**0.63**
wikiSERA-10	**0.706**	**0.733**	**0.793**	0.822	**0.823**	**0.847**	0.845	0.824	**0.652**	**0.715**	0.769	0.791	0.781	0.798	0.799	0.798	**0.462**	**0.516**	0.565	0.591	**0.601**	0.599	0.604	0.598
wikiSERA-DIS-5	0.571	0.645	0.742	0.811	0.809	0.819	0.856	0.828	0.443	0.585	0.701	0.782	**0.783**	**0.82**	0.825	0.8	0.315	0.421	0.511	0.59	0.589	**0.626**	0.624	0.602
wikiSERA-DIS-10	0.677	0.722	0.788	**0.827**	0.819	0.829	0.851	0.827	0.619	0.697	**0.775**	**0.814**	**0.783**	0.806	0.8	0.798	0.436	0.496	**0.587**	**0.616**	0.599	0.603	0.608	0.598

Table A.16: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.526	0.641	0.697	0.821	0.781	**0.843**	0.835	0.814	0.369	0.572	0.68	0.786	0.761	0.793	0.816	0.8	0.269	0.414	0.488	0.6	0.563	**0.593**	0.615	0.599
SERA-10	**0.72**	0.709	0.8	0.819	0.82	0.842	0.808	0.803	0.664	0.655	0.77	0.789	0.782	0.796	0.766	0.774	0.476	0.479	0.575	0.59	0.585	0.584	0.57	0.563
SERA-NP-5	0.56	0.668	0.694	0.767	0.785	0.81	0.788	0.842	0.45	0.615	0.685	0.757	0.753	0.762	0.76	**0.841**	0.319	0.441	0.485	0.563	0.549	0.562	0.558	**0.646**
SERA-NP-10	0.711	**0.718**	0.792	**0.828**	**0.833**	0.824	**0.845**	0.812	0.66	**0.665**	**0.775**	**0.806**	**0.814**	0.76	**0.823**	0.793	0.474	**0.492**	**0.579**	**0.61**	**0.617**	0.562	**0.627**	0.594
SERA-KW-5	0.518	0.639	0.693	0.821	0.779	0.837	0.829	0.821	0.365	0.577	0.676	0.777	0.759	0.786	0.799	0.764	0.264	0.423	0.484	0.587	0.562	0.577	0.602	0.565
SERA-KW-10	0.716	0.706	**0.804**	0.816	0.816	0.837	0.815	**0.85**	**0.67**	0.662	0.77	0.78	0.779	0.793	0.78	0.804	**0.482**	0.482	0.574	0.576	0.584	0.579	0.584	0.61
SERA-DIS-5	0.495	0.616	0.709	0.815	0.77	0.811	0.831	0.825	0.34	0.556	0.688	0.776	0.742	0.77	0.8	0.81	0.238	0.404	0.489	0.587	0.533	0.567	0.594	0.609
SERA-DIS-10	0.652	0.687	0.786	0.808	0.813	0.832	0.821	0.815	0.595	0.661	0.757	0.782	0.785	**0.797**	0.792	0.788	0.418	0.478	0.559	0.587	0.579	0.591	0.585	0.576
SERA-DIS-NP-5	0.554	0.644	0.671	0.77	0.786	0.785	0.793	0.835	0.44	0.588	0.677	0.747	0.734	0.751	0.767	0.8	0.305	0.421	0.487	0.561	0.537	0.547	0.569	0.597
SERA-DIS-NP-10	0.648	0.683	0.759	0.814	0.828	0.819	0.832	0.825	0.599	0.628	0.744	0.8	0.795	0.78	0.802	0.785	0.419	0.455	0.547	0.607	0.599	0.574	0.602	0.577
SERA-DIS-KW-5	0.477	0.62	0.708	0.81	0.766	0.807	0.827	0.822	0.311	0.57	0.694	0.777	0.744	0.775	0.79	0.768	0.226	0.416	0.487	0.582	0.542	0.571	0.577	0.569
SERA-DIS-KW-10	0.648	0.692	0.79	0.8	0.81	0.827	0.824	0.837	0.585	0.663	0.765	0.779	0.783	0.792	0.796	0.78	0.408	0.482	0.561	0.576	0.575	0.586	0.589	0.58
wikiSERA-5	0.55	0.638	0.732	0.818	0.8	0.836	**0.857**	**0.838**	0.414	0.596	0.689	0.772	0.772	**0.827**	**0.832**	**0.822**	0.286	0.432	0.501	0.581	0.562	**0.636**	**0.627**	**0.625**
wikiSERA-10	**0.697**	**0.708**	**0.795**	**0.831**	**0.82**	**0.842**	0.835	0.831	**0.627**	**0.659**	**0.748**	0.802	**0.781**	0.795	0.794	0.809	**0.442**	**0.49**	**0.553**	0.605	**0.588**	0.593	0.597	0.61
wikiSERA-DIS-5	0.556	0.63	0.719	0.807	0.779	0.815	0.853	0.829	0.444	0.601	0.671	0.784	0.741	0.809	0.821	0.795	0.311	0.434	0.487	0.588	0.545	0.615	0.622	0.599
wikiSERA-DIS-10	0.658	0.686	0.776	0.829	0.804	0.829	0.847	0.834	0.604	0.652	0.728	**0.809**	0.772	0.805	0.806	0.802	0.428	0.479	0.541	**0.61**	0.58	0.601	0.608	0.604

Table A.17: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

	Pearson								Spearman								Kendall							
Method	TAC2008	AQUAINT-2							TAC2008	AQUAINT-2							TAC2008	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.532	0.644	0.705	0.814	0.791	**0.847**	0.845	0.818	0.373	0.574	0.692	0.787	0.765	0.799	0.821	0.804	0.264	0.419	0.507	0.604	0.564	**0.601**	0.624	0.606
SERA-10	**0.716**	0.716	0.794	0.819	0.824	0.844	0.817	0.812	**0.653**	0.674	0.775	0.796	0.792	0.792	0.766	0.784	**0.47**	0.488	0.575	0.598	0.6	0.578	0.572	0.576
SERA-NP-5	0.566	0.689	0.693	0.781	0.798	0.815	0.792	0.833	0.452	0.661	0.708	0.771	0.756	0.762	0.77	**0.827**	0.326	0.484	0.506	0.582	0.558	0.566	0.569	**0.635**
SERA-NP-10	0.703	**0.734**	**0.798**	**0.827**	**0.835**	0.827	**0.849**	0.81	0.646	**0.689**	**0.787**	0.8	**0.816**	0.772	**0.827**	0.785	0.462	**0.505**	**0.586**	0.601	**0.621**	0.568	**0.632**	0.582
SERA-KW-5	0.527	0.643	0.708	0.818	0.789	0.841	0.841	0.823	0.36	0.581	0.702	0.787	0.767	0.796	0.809	0.763	0.26	0.426	0.518	**0.605**	0.564	0.6	0.611	0.566
SERA-KW-10	0.71	0.715	0.797	0.815	0.821	0.84	0.823	**0.84**	0.651	0.675	0.778	0.793	0.789	0.788	0.779	0.799	0.465	0.487	0.579	0.59	0.598	0.569	0.579	0.611
SERA-DIS-5	0.513	0.632	0.708	0.809	0.781	0.821	0.84	0.828	0.344	0.572	0.696	0.786	0.758	0.788	0.8	0.809	0.241	0.412	0.507	0.596	0.556	0.592	0.593	0.605
SERA-DIS-10	0.662	0.701	0.783	0.808	0.814	0.836	0.829	0.824	0.583	0.67	0.761	0.795	0.795	**0.8**	0.797	0.796	0.414	0.48	0.565	0.603	0.596	0.596	0.592	0.592
SERA-DIS-NP-5	0.555	0.666	0.682	0.789	0.795	0.799	0.805	0.824	0.438	0.626	0.699	0.767	0.754	0.753	0.774	0.777	0.308	0.448	0.5	0.581	0.557	0.555	0.573	0.588
SERA-DIS-NP-10	0.644	0.708	0.77	0.821	0.833	0.824	0.843	0.817	0.588	0.672	0.764	**0.801**	0.79	0.784	0.819	0.778	0.415	0.485	0.564	0.603	0.593	0.578	0.616	0.576
SERA-DIS-KW-5	0.5	0.636	0.712	0.808	0.778	0.815	0.837	0.826	0.337	0.581	0.702	0.787	0.748	0.788	0.801	0.779	0.241	0.422	0.503	0.599	0.55	0.592	0.597	0.579
SERA-DIS-KW-10	0.657	0.708	0.788	0.801	0.812	0.831	0.832	0.835	0.583	0.682	0.766	0.784	0.791	0.797	0.804	0.789	0.412	0.493	0.57	0.588	0.586	0.596	0.601	0.594
wikiSERA-5	0.557	0.642	0.742	0.814	0.803	0.839	**0.869**	**0.846**	0.424	0.592	0.703	0.78	0.779	**0.822**	**0.854**	**0.833**	0.293	0.431	0.519	0.591	0.581	**0.632**	**0.663**	**0.641**
wikiSERA-10	**0.707**	**0.72**	**0.795**	0.829	**0.825**	**0.846**	0.843	0.828	**0.646**	**0.693**	**0.767**	0.801	**0.783**	0.804	0.8	0.804	**0.457**	**0.502**	**0.569**	0.603	**0.599**	0.603	0.604	0.605
wikiSERA-DIS-5	0.565	0.629	0.733	0.809	0.792	0.824	0.861	0.839	0.456	0.599	0.7	0.785	0.764	0.814	0.836	0.806	0.323	0.434	0.517	0.595	0.57	0.619	0.641	0.608
wikiSERA-DIS-10	0.673	0.704	0.786	**0.83**	0.814	0.835	0.852	0.837	0.614	0.687	0.746	**0.816**	0.782	0.808	0.806	0.807	0.434	**0.502**	0.556	**0.619**	0.586	0.605	0.61	0.608

Table A.18: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.3 Correlation of SERA and wikiSERA with Pyramid on TAC2009/AQUAINT-2

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.849	0.904	0.895	0.863	0.837	0.843	0.857	0.866	0.697	0.818	0.833	0.8	0.812	0.832	0.833	0.796	0.524	0.635	0.664	0.622	0.64	0.668	0.657	0.624
	SERA-10	0.771	0.9	0.887	0.851	0.822	0.858	0.845	0.859	0.674	0.818	0.856	0.852	0.778	0.871	0.818	0.797	0.521	0.646	0.691	**0.688**	0.599	0.699	0.648	0.626
	SERA-NP-5	0.875	0.917	0.914	0.855	0.826	0.795	0.848	0.85	0.766	0.81	0.852	0.779	0.752	0.776	0.824	0.793	0.58	0.632	0.689	0.619	0.582	0.609	0.665	0.617
	SERA-NP-10	0.8	0.905	0.888	0.821	0.844	0.842	0.843	0.859	0.745	0.808	0.836	0.81	0.82	0.827	0.838	0.816	0.571	0.635	0.679	0.647	0.641	0.657	0.673	0.639
	SERA-KW-5	0.838	0.896	0.886	0.86	0.828	0.839	0.857	0.861	0.68	0.808	0.826	0.792	0.807	0.817	0.832	0.791	0.503	0.626	0.657	0.614	0.635	0.652	0.659	0.62
	SERA-KW-10	0.771	0.899	0.88	0.852	0.814	0.853	0.844	0.856	0.672	0.811	0.848	**0.855**	0.776	0.862	0.815	0.792	0.518	0.634	0.685	0.683	0.601	0.691	0.64	0.629
	SERA-DIS-5	0.87	0.924	0.933	0.906	0.895	0.901	0.893	0.908	0.696	0.809	0.865	0.804	0.825	0.847	0.842	0.836	0.511	0.626	0.692	0.636	0.654	0.681	0.67	0.665
	SERA-DIS-10	0.894	0.935	**0.946**	**0.924**	0.91	**0.927**	0.913	0.932	0.719	0.813	**0.878**	0.835	**0.834**	**0.906**	0.847	0.852	0.554	0.642	**0.712**	0.668	0.661	**0.754**	0.678	0.678
	SERA-DIS-NP-5	0.89	0.932	0.939	0.908	0.879	0.882	0.898	0.903	0.745	**0.828**	0.832	0.827	0.728	0.806	0.858	0.848	0.57	0.643	0.662	0.67	0.557	0.628	0.7	0.689
	SERA-DIS-NP-10	**0.906**	0.935	0.944	0.912	**0.911**	0.92	**0.924**	**0.94**	**0.774**	0.823	0.844	0.81	0.809	0.819	**0.873**	**0.873**	**0.62**	0.647	0.689	0.647	0.635	0.643	**0.721**	**0.702**
	SERA-DIS-KW-5	0.863	0.918	0.93	0.903	0.886	0.894	0.893	0.899	0.664	0.8	0.853	0.794	0.83	0.82	0.85	0.825	0.486	0.615	0.682	0.624	**0.665**	0.652	0.675	0.654
	SERA-DIS-KW-10	0.891	**0.936**	0.94	0.923	0.902	**0.927**	0.911	0.929	0.708	0.826	0.866	0.833	0.819	0.888	0.842	0.835	0.53	**0.653**	0.694	0.666	0.654	0.732	0.667	0.666
	wikiSERA-5	0.84	0.905	0.904	0.853	0.823	0.834	0.856	0.861	0.719	0.808	0.862	0.802	0.745	0.847	0.827	0.803	0.551	0.636	0.704	0.626	0.576	0.68	0.675	0.62
	wikiSERA-10	0.788	0.905	0.884	0.838	0.813	0.842	0.843	0.855	0.667	0.817	0.858	0.859	0.762	0.874	0.811	0.812	0.514	0.64	0.698	0.695	0.587	0.699	0.637	0.644
	wikiSERA-DIS-5	0.882	0.922	0.944	0.911	0.881	0.891	0.891	0.906	**0.753**	0.808	0.862	0.83	0.779	0.864	0.862	0.859	0.576	0.639	0.704	0.666	0.609	0.69	0.696	0.682
	wikiSERA-DIS-10	**0.908**	**0.939**	**0.949**	**0.924**	**0.897**	**0.92**	**0.916**	**0.932**	0.746	**0.832**	**0.894**	**0.884**	**0.808**	**0.886**	**0.863**	**0.862**	**0.584**	**0.651**	**0.743**	**0.719**	**0.635**	**0.728**	**0.697**	**0.688**

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.848	0.904	0.894	0.863	0.837	0.842	0.857	0.866	0.7	0.818	0.835	0.802	0.816	0.832	0.833	0.797	0.527	0.635	0.668	0.627	0.643	0.668	0.66	0.627
	SERA-10	0.77	0.9	0.887	0.851	0.821	0.858	0.845	0.858	0.676	0.821	0.859	0.852	0.781	0.872	0.819	0.799	0.522	0.651	0.696	**0.688**	0.599	0.702	0.647	0.631
	SERA-NP-5	0.874	0.917	0.914	0.855	0.825	0.795	0.848	0.85	0.768	0.812	0.855	0.781	0.756	0.775	0.825	0.793	0.582	0.635	0.69	0.621	0.584	0.607	0.67	0.616
	SERA-NP-10	0.799	0.903	0.888	0.821	0.844	0.842	0.843	0.858	0.748	0.813	0.84	0.812	0.824	0.828	0.838	0.816	0.574	0.641	0.681	0.649	0.643	0.657	0.673	0.641
	SERA-KW-5	0.838	0.896	0.886	0.86	0.827	0.838	0.857	0.86	0.683	0.809	0.828	0.793	0.811	0.817	0.833	0.792	0.505	0.626	0.66	0.619	0.637	0.651	0.661	0.622
	SERA-KW-10	0.77	0.899	0.88	0.851	0.813	0.852	0.844	0.855	0.674	0.814	0.851	**0.855**	0.778	0.863	0.816	0.793	0.518	0.639	0.69	0.684	0.601	0.694	0.64	0.633
	SERA-DIS-5	0.87	0.924	0.933	0.906	0.894	0.9	0.893	0.908	0.698	0.809	0.867	0.806	0.828	0.848	0.844	0.838	0.513	0.624	0.695	0.641	0.659	0.683	0.668	0.667
	SERA-DIS-10	0.894	0.935	**0.946**	**0.924**	0.909	**0.927**	0.913	0.931	0.721	0.814	**0.879**	0.836	**0.836**	**0.906**	0.848	0.855	0.555	0.644	**0.715**	0.667	0.66	**0.756**	0.678	0.683
	SERA-DIS-NP-5	0.89	0.932	0.939	0.908	0.878	0.881	0.898	0.903	0.747	**0.829**	0.835	0.829	0.731	0.807	0.86	0.849	0.572	0.647	0.666	0.672	0.559	0.632	0.705	0.694
	SERA-DIS-NP-10	**0.905**	0.934	0.944	0.911	**0.91**	0.92	**0.923**	**0.94**	**0.776**	0.821	0.847	0.812	0.813	0.82	**0.874**	**0.874**	**0.621**	0.653	0.694	0.649	0.637	0.644	**0.723**	**0.703**
	SERA-DIS-KW-5	0.862	0.918	0.93	0.903	0.885	0.894	0.893	0.899	0.667	0.801	0.855	0.796	0.834	0.821	0.85	0.827	0.488	0.613	0.686	0.629	**0.67**	0.652	0.674	0.656
	SERA-DIS-KW-10	0.89	**0.936**	0.939	0.922	0.902	**0.927**	0.911	0.929	0.709	0.828	0.867	0.834	0.822	0.889	0.842	0.838	0.531	**0.655**	0.698	0.666	0.653	0.734	0.667	0.671
	wikiSERA-5	0.84	0.905	0.904	0.852	0.822	0.833	0.855	0.861	0.72	0.807	0.864	0.804	0.747	0.848	0.829	0.805	0.552	0.634	0.706	0.629	0.579	0.68	0.676	0.623
	wikiSERA-10	0.787	0.905	0.883	0.837	0.812	0.841	0.843	0.855	0.669	0.82	0.861	0.86	0.765	0.876	0.812	0.813	0.516	0.645	0.703	0.697	0.587	0.701	0.637	0.649
	wikiSERA-DIS-5	0.882	0.921	0.944	0.911	0.88	0.891	0.891	0.906	**0.755**	0.806	0.864	0.833	0.78	0.866	0.864	0.861	0.577	0.639	0.707	0.668	0.612	0.694	**0.699**	0.686
	wikiSERA-DIS-10	**0.907**	**0.939**	**0.949**	**0.924**	**0.897**	**0.92**	**0.916**	**0.932**	0.748	**0.834**	**0.896**	**0.885**	**0.81**	**0.886**	**0.865**	**0.864**	**0.586**	**0.653**	**0.748**	**0.721**	**0.634**	**0.732**	**0.699**	**0.692**

Table A.19: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.889	0.886	0.901	0.88	0.876	0.834	0.862	0.874	0.746	0.766	0.792	0.805	**0.882**	0.746	0.746	0.775	0.587	0.585	0.629	0.638	**0.732**	0.564	0.578	0.6
	SERA-10	0.845	0.905	0.879	0.865	0.858	0.868	0.873	0.888	**0.805**	**0.822**	0.802	0.83	0.846	**0.85**	0.802	0.809	**0.639**	**0.651**	**0.649**	0.662	0.671	**0.681**	0.634	0.644
	SERA-NP-5	0.877	0.883	0.91	0.867	0.85	0.86	0.851	0.863	0.744	0.757	0.804	0.756	0.829	0.818	0.791	0.78	0.578	0.596	0.643	0.588	0.651	0.641	0.624	0.604
	SERA-NP-10	0.85	0.881	0.886	0.866	0.856	0.862	0.869	0.882	0.796	0.809	**0.808**	0.8	0.821	0.847	**0.813**	**0.829**	0.619	0.626	0.646	0.627	0.648	0.666	**0.64**	**0.655**
	SERA-KW-5	0.883	0.879	0.895	0.878	0.867	0.83	0.848	0.871	0.74	0.761	0.781	0.798	0.873	0.741	0.735	0.752	0.579	0.58	0.624	0.633	0.724	0.562	0.567	0.58
	SERA-KW-10	0.837	0.905	0.879	0.865	0.852	0.864	0.868	0.887	0.776	0.815	0.792	0.828	0.831	0.841	0.784	0.812	0.604	0.642	0.636	0.655	0.656	0.668	0.607	0.644
	SERA-DIS-5	0.893	0.893	0.928	0.924	0.917	0.884	0.894	0.89	0.698	0.757	0.783	0.825	0.869	0.732	0.765	0.734	0.538	0.561	0.616	0.666	0.72	0.55	0.599	0.566
	SERA-DIS-10	**0.927**	**0.928**	0.929	**0.937**	**0.93**	**0.925**	0.915	**0.926**	0.779	0.811	0.793	**0.845**	0.857	0.829	0.787	0.814	0.619	0.627	0.634	**0.674**	0.696	0.654	0.613	0.642
	SERA-DIS-NP-5	0.89	0.888	0.93	0.91	0.908	0.906	0.889	0.893	0.708	0.732	0.807	0.746	0.832	0.811	0.762	0.755	0.547	0.561	0.644	0.585	0.658	0.636	0.581	0.582
	SERA-DIS-NP-10	0.916	0.92	**0.939**	0.93	0.923	0.919	**0.921**	0.917	0.737	0.797	0.794	0.79	0.832	0.814	0.8	0.81	0.572	0.615	0.638	0.616	0.658	0.631	0.636	0.635
	SERA-DIS-KW-5	0.891	0.885	0.925	0.919	0.909	0.882	0.883	0.885	0.724	0.736	0.772	0.818	0.862	0.721	0.742	0.719	0.554	0.542	0.603	0.665	0.706	0.539	0.566	0.558
	SERA-DIS-KW-10	0.914	0.926	0.927	0.933	0.923	0.92	0.907	0.915	0.753	0.805	0.78	0.83	0.863	0.815	0.761	0.792	0.595	0.622	0.624	0.663	0.702	0.64	0.589	0.626
	wikiSERA-5	0.874	0.891	0.906	0.887	0.863	0.828	0.841	0.862	0.724	0.779	**0.813**	0.808	**0.84**	0.751	0.726	0.769	0.557	0.594	**0.663**	0.642	**0.694**	0.568	0.556	0.591
	wikiSERA-10	0.843	0.905	0.872	0.864	0.849	0.85	0.866	0.892	**0.796**	**0.802**	0.786	**0.833**	0.823	**0.822**	**0.796**	**0.838**	0.625	**0.623**	0.625	**0.666**	0.644	**0.643**	**0.621**	**0.672**
	wikiSERA-DIS-5	0.898	0.899	**0.933**	0.925	0.901	0.887	0.885	0.886	0.703	0.775	0.78	0.796	0.806	0.773	0.723	0.748	0.537	0.594	0.62	0.63	0.646	0.595	0.547	0.566
	wikiSERA-DIS-10	**0.923**	**0.923**	0.927	**0.931**	**0.921**	**0.918**	**0.909**	**0.926**	0.794	0.79	0.778	0.819	0.827	0.805	0.76	0.817	**0.634**	0.614	0.612	0.648	0.65	0.622	0.582	0.651

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.889	0.886	0.901	0.879	0.875	0.833	0.862	0.874	0.749	0.769	0.795	0.807	**0.884**	0.747	0.749	0.778	0.59	0.587	0.634	0.641	**0.734**	0.566	0.58	0.602
	SERA-10	0.844	0.905	0.878	0.864	0.858	0.868	0.873	0.888	**0.807**	**0.825**	0.805	0.831	0.848	**0.852**	0.802	0.809	**0.641**	**0.653**	**0.652**	0.665	0.674	**0.683**	0.636	0.646
	SERA-NP-5	0.877	0.883	0.91	0.866	0.85	0.859	0.851	0.863	0.747	0.759	0.809	0.758	0.832	0.817	0.792	0.781	0.58	0.597	0.648	0.591	0.656	0.641	0.624	0.604
	SERA-NP-10	0.85	0.881	0.886	0.866	0.855	0.861	0.868	0.882	0.798	0.808	**0.812**	0.8	0.822	0.848	**0.813**	**0.829**	0.621	0.631	0.651	0.626	0.648	0.666	**0.64**	**0.655**
	SERA-KW-5	0.883	0.88	0.895	0.877	0.867	0.829	0.848	0.871	0.744	0.765	0.784	0.8	0.875	0.743	0.738	0.754	0.581	0.582	0.629	0.635	0.726	0.565	0.569	0.582
	SERA-KW-10	0.836	0.905	0.879	0.865	0.851	0.863	0.868	0.887	0.777	0.819	0.795	0.829	0.833	0.843	0.785	0.813	0.606	0.644	0.638	0.657	0.658	0.67	0.61	0.647
	SERA-DIS-5	0.893	0.894	0.929	0.924	0.917	0.884	0.894	0.89	0.701	0.761	0.785	0.827	0.872	0.733	0.768	0.736	0.54	0.563	0.621	0.668	0.722	0.552	0.601	0.569
	SERA-DIS-10	**0.927**	**0.928**	0.929	**0.937**	**0.93**	**0.925**	0.915	**0.926**	0.782	0.814	0.795	**0.846**	0.86	0.831	0.789	0.815	0.621	0.629	0.636	**0.676**	0.698	0.656	0.616	0.641
	SERA-DIS-NP-5	0.89	0.888	0.93	0.91	0.908	0.905	0.889	0.893	0.71	0.735	0.81	0.747	0.835	0.811	0.763	0.756	0.55	0.563	0.649	0.587	0.663	0.636	0.581	0.585
	SERA-DIS-NP-10	0.916	0.925	**0.939**	0.93	0.923	0.919	**0.921**	0.918	0.74	0.804	0.798	0.791	0.834	0.815	0.799	0.811	0.574	0.616	0.643	0.616	0.657	0.631	0.636	0.635
	SERA-DIS-KW-5	0.891	0.885	0.926	0.92	0.909	0.882	0.883	0.886	0.727	0.74	0.774	0.82	0.866	0.722	0.745	0.721	0.556	0.544	0.608	0.667	0.709	0.542	0.568	0.56
	SERA-DIS-KW-10	0.914	0.926	0.927	0.933	0.923	0.92	0.907	0.916	0.755	0.808	0.782	0.832	0.866	0.817	0.763	0.793	0.597	0.624	0.626	0.666	0.705	0.643	0.591	0.625
	wikiSERA-5	0.874	0.892	0.906	0.887	0.863	0.827	0.841	0.862	0.727	0.781	**0.815**	0.81	**0.844**	0.753	0.728	0.771	0.559	0.596	**0.666**	0.644	**0.697**	0.57	0.558	0.593
	wikiSERA-10	0.843	0.905	0.872	0.864	0.849	0.849	0.865	0.892	**0.798**	**0.804**	0.789	**0.834**	0.824	**0.824**	**0.796**	**0.838**	0.627	**0.625**	0.628	**0.668**	0.644	**0.645**	**0.62**	**0.672**
	wikiSERA-DIS-5	0.898	0.9	**0.934**	0.925	0.901	0.887	0.884	0.886	0.707	0.777	0.782	0.797	0.808	0.774	0.725	0.75	0.539	0.596	0.625	0.632	0.648	0.597	0.55	0.567
	wikiSERA-DIS-10	**0.923**	**0.924**	0.927	**0.931**	**0.921**	**0.918**	**0.909**	**0.926**	0.797	0.792	0.78	0.821	0.828	0.806	0.761	0.817	**0.636**	0.617	0.614	0.651	0.649	0.624	0.585	0.651

Table A.20: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.794	0.864	0.858	0.813	0.802	0.816	0.843	0.851	0.519	0.638	0.743	0.749	0.733	0.764	0.764	0.757	0.383	0.488	0.571	0.569	0.563	0.582	0.598	0.582
	SERA-10	0.784	0.882	0.845	0.803	0.81	0.85	0.858	0.879	0.679	0.714	0.776	0.776	0.79	**0.789**	0.784	0.816	0.512	0.542	0.604	0.598	0.625	**0.624**	0.611	0.655
	SERA-NP-5	0.806	0.852	0.881	0.841	0.842	0.808	0.832	0.846	0.599	0.625	0.775	0.749	0.771	0.787	0.798	0.746	0.437	0.465	0.622	0.581	0.597	0.618	0.643	0.574
	SERA-NP-10	0.806	0.875	0.854	0.808	0.811	0.845	0.859	0.871	0.687	0.659	0.763	0.759	0.739	0.783	0.807	0.769	0.519	0.534	0.593	0.589	0.565	0.615	0.646	0.607
	SERA-KW-5	0.803	0.856	0.858	0.821	0.802	0.821	0.845	0.851	0.537	0.623	0.765	0.748	0.726	0.759	0.763	0.765	0.396	0.466	0.591	0.57	0.555	0.569	0.601	0.584
	SERA-KW-10	0.785	0.885	0.847	0.805	0.81	0.847	0.859	0.882	**0.691**	**0.723**	0.756	**0.787**	0.789	0.782	0.785	0.823	**0.525**	**0.549**	0.585	**0.613**	0.625	0.612	0.616	0.665
	SERA-DIS-5	0.802	0.874	0.908	0.882	0.871	0.875	0.891	0.896	0.444	0.621	0.766	0.754	0.791	0.75	**0.814**	0.789	0.316	0.463	0.584	0.57	0.615	0.566	0.635	0.605
	SERA-DIS-10	0.879	0.907	0.92	0.897	0.902	**0.916**	0.918	**0.934**	0.586	0.707	0.79	0.784	**0.821**	0.781	0.778	0.843	0.434	0.533	0.609	0.608	**0.647**	0.608	0.612	**0.682**
	SERA-DIS-NP-5	0.849	0.871	0.924	0.898	**0.905**	0.867	0.888	0.888	0.528	0.601	0.778	0.735	0.797	0.756	0.804	0.756	0.397	0.448	0.623	0.566	0.626	0.577	0.627	0.584
	SERA-DIS-NP-10	**0.895**	0.894	**0.933**	0.9	0.904	0.909	**0.923**	0.92	0.596	0.715	**0.796**	0.744	0.767	0.784	0.812	0.777	0.448	0.54	**0.636**	0.568	0.592	0.607	**0.647**	0.604
	SERA-DIS-KW-5	0.811	0.872	0.908	0.887	0.869	0.869	0.892	0.899	0.457	0.609	0.763	0.736	0.77	0.711	0.799	0.787	0.333	0.449	0.584	0.554	0.593	0.534	0.623	0.603
	SERA-DIS-KW-10	0.881	**0.91**	0.92	**0.901**	0.898	0.911	0.92	0.93	0.622	0.716	0.789	0.779	0.813	0.752	0.782	**0.846**	0.461	0.541	0.603	0.608	0.635	0.575	0.614	0.678
	wikiSERA-5	0.79	0.854	0.863	0.822	0.816	0.809	0.849	0.846	0.519	0.597	**0.81**	0.773	0.717	0.702	0.799	0.724	0.376	0.436	**0.646**	0.592	0.549	0.529	0.629	0.558
	wikiSERA-10	0.77	0.877	0.832	0.806	0.799	0.846	0.85	0.887	**0.673**	**0.736**	0.763	**0.804**	**0.773**	0.788	0.77	**0.821**	**0.501**	**0.556**	0.596	**0.633**	**0.602**	0.625	0.603	**0.66**
	wikiSERA-DIS-5	0.833	0.865	**0.912**	0.883	0.875	0.876	0.894	0.884	0.521	0.581	0.783	0.746	0.766	0.726	**0.834**	0.724	0.367	0.434	0.612	0.576	0.587	0.552	**0.665**	0.549
	wikiSERA-DIS-10	**0.881**	**0.902**	**0.912**	**0.895**	**0.893**	**0.915**	**0.916**	**0.932**	0.632	0.706	0.762	0.777	**0.773**	**0.806**	0.801	0.804	0.471	0.531	0.592	0.595	0.584	**0.641**	0.631	0.63

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.793	0.864	0.858	0.813	0.801	0.815	0.843	0.851	0.519	0.639	0.747	0.754	0.737	0.765	0.766	0.76	0.383	0.488	0.575	0.571	0.565	0.584	0.603	0.584
	SERA-10	0.783	0.882	0.845	0.802	0.809	0.849	0.858	0.879	0.681	0.716	0.78	0.778	0.793	**0.788**	0.788	0.819	0.514	0.544	0.609	0.6	0.625	**0.622**	0.616	0.657
	SERA-NP-5	0.805	0.852	0.88	0.84	0.842	0.807	0.832	0.846	0.599	0.626	0.777	0.751	0.774	0.787	0.797	0.748	0.438	0.466	0.624	0.583	0.599	0.618	0.64	0.576
	SERA-NP-10	0.805	0.87	0.854	0.807	0.811	0.845	0.859	0.871	0.689	0.671	0.766	0.76	0.742	0.783	0.807	0.771	0.521	0.508	0.595	0.591	0.567	0.617	0.645	0.609
	SERA-KW-5	0.803	0.857	0.858	0.82	0.801	0.82	0.844	0.85	0.536	0.624	0.768	0.753	0.73	0.76	0.764	0.769	0.397	0.467	0.595	0.572	0.557	0.57	0.606	0.587
	SERA-KW-10	0.784	0.885	0.846	0.804	0.809	0.847	0.859	0.881	**0.694**	**0.724**	0.761	**0.79**	0.793	0.781	0.789	0.826	**0.527**	**0.551**	0.59	**0.615**	0.627	0.611	0.621	0.67
	SERA-DIS-5	0.802	0.874	0.907	0.882	0.871	0.875	0.891	0.896	0.443	0.621	0.767	0.759	0.795	0.751	**0.815**	0.792	0.317	0.462	0.587	0.573	0.62	0.569	0.635	0.61
	SERA-DIS-10	0.879	0.907	0.92	0.896	0.901	**0.916**	0.918	**0.934**	0.587	0.708	0.793	0.787	**0.824**	0.781	0.782	0.847	0.437	0.532	0.614	0.61	**0.649**	0.608	0.614	**0.687**
	SERA-DIS-NP-5	0.849	0.871	0.924	0.897	**0.905**	0.867	0.887	0.888	0.526	0.6	0.778	0.737	0.799	0.757	0.804	0.758	0.395	0.449	0.621	0.571	0.628	0.577	0.624	0.586
	SERA-DIS-NP-10	**0.895**	0.897	**0.933**	**0.9**	0.904	0.909	**0.923**	0.92	0.598	0.715	**0.798**	0.745	0.769	0.785	0.813	0.779	0.449	0.541	**0.637**	0.57	0.594	0.609	**0.647**	0.606
	SERA-DIS-KW-5	0.811	0.873	0.907	0.887	0.869	0.868	0.892	0.899	0.456	0.609	0.764	0.741	0.774	0.712	0.799	0.789	0.334	0.449	0.587	0.556	0.598	0.534	0.622	0.608
	SERA-DIS-KW-10	0.881	**0.91**	0.919	**0.9**	0.898	0.911	0.919	0.931	0.622	0.717	0.792	0.782	0.816	0.751	0.785	**0.849**	0.463	0.54	0.608	0.61	0.637	0.575	0.617	0.683
	wikiSERA-5	0.79	0.854	0.863	0.821	0.815	0.809	0.848	0.846	0.519	0.596	**0.813**	0.776	0.721	0.704	0.799	0.728	0.377	0.434	**0.647**	0.594	0.553	0.528	0.628	0.562
	wikiSERA-10	0.769	0.877	0.832	0.806	0.798	0.845	0.85	0.887	**0.675**	**0.739**	0.767	**0.806**	0.776	0.787	0.773	**0.824**	**0.503**	**0.558**	0.598	**0.635**	**0.604**	0.624	0.605	**0.665**
	wikiSERA-DIS-5	0.833	0.865	**0.912**	0.882	0.875	0.876	0.894	0.884	0.52	0.581	0.785	0.75	0.77	0.729	**0.835**	0.727	0.365	0.434	0.613	0.578	0.589	0.554	**0.664**	0.554
	wikiSERA-DIS-10	**0.881**	**0.902**	**0.912**	**0.895**	**0.892**	**0.915**	**0.916**	**0.932**	0.632	0.707	0.766	0.779	**0.777**	**0.807**	0.805	0.808	0.469	0.533	0.595	0.597	0.589	**0.644**	0.633	0.635

Table A.21: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_3$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.861	0.885	0.906	0.871	0.874	0.859	0.892	0.908	0.714	0.748	0.798	0.797	0.811	0.762	0.746	0.776	0.548	0.576	0.641	0.624	0.635	0.59	0.57	0.609
	SERA-10	0.829	0.904	0.891	0.854	0.87	0.881	0.914	**0.931**	0.775	0.773	**0.821**	0.808	0.795	0.826	0.827	0.816	0.6	0.602	**0.657**	0.639	0.632	0.655	**0.66**	0.645
	SERA-NP-5	0.849	0.888	0.912	0.872	0.852	0.867	0.882	0.893	0.766	0.745	0.773	0.761	0.765	0.81	0.813	0.753	0.597	0.574	0.608	0.6	0.6	0.646	0.656	0.588
	SERA-NP-10	0.836	0.891	0.89	0.859	0.845	0.868	0.892	0.913	**0.776**	0.746	0.812	0.785	0.715	0.801	0.796	0.809	**0.603**	0.577	0.649	0.616	0.541	0.633	0.626	0.646
	SERA-KW-5	0.863	0.885	0.902	0.87	0.872	0.853	0.892	0.904	0.73	0.745	0.783	0.797	0.802	0.747	0.774	0.784	0.563	0.576	0.633	0.626	0.626	0.573	0.599	0.61
	SERA-KW-10	0.828	0.908	0.89	0.854	0.868	0.877	0.916	0.93	0.762	**0.78**	0.819	0.805	0.783	0.82	**0.83**	0.817	0.588	**0.613**	**0.657**	0.639	0.618	0.658	0.656	0.646
	SERA-DIS-5	0.89	0.895	0.931	0.912	0.927	0.896	0.916	0.928	0.696	0.731	0.775	0.76	**0.826**	0.766	0.705	0.798	0.533	0.565	0.608	0.581	**0.659**	0.592	0.541	0.623
	SERA-DIS-10	**0.907**	**0.911**	0.939	0.918	**0.936**	0.924	**0.927**	0.928	0.717	0.733	0.799	0.778	0.807	0.822	0.784	0.823	0.552	0.558	0.638	0.595	0.644	0.655	0.609	0.643
	SERA-DIS-NP-5	0.876	0.897	0.934	0.931	0.915	0.927	0.908	0.921	0.748	0.728	0.775	0.784	0.791	**0.858**	0.787	0.769	0.569	0.561	0.605	0.614	0.62	0.684	0.627	0.6
	SERA-DIS-NP-10	0.897	0.902	**0.945**	**0.937**	0.923	**0.937**	0.921	0.928	0.746	0.741	0.811	**0.823**	0.753	0.852	0.789	0.799	0.574	0.577	0.647	**0.654**	0.591	**0.688**	0.624	0.643
	SERA-DIS-KW-5	0.897	0.896	0.929	0.912	0.922	0.891	0.915	0.923	0.708	0.731	0.785	0.783	0.808	0.764	0.748	0.811	0.547	0.566	0.619	0.603	0.639	0.58	0.574	0.634
	SERA-DIS-KW-10	0.906	0.91	0.939	0.918	0.93	0.919	**0.927**	0.923	0.721	0.733	0.805	0.784	0.791	0.821	0.786	**0.827**	0.555	0.557	0.638	0.601	0.624	0.655	0.615	**0.655**
	wikiSERA-5	0.834	0.885	0.903	0.878	0.868	0.837	0.887	0.913	0.675	0.756	0.781	0.788	0.785	0.74	0.791	0.815	0.51	0.577	0.622	0.619	0.624	0.564	0.62	0.643
	wikiSERA-10	0.848	0.911	0.885	0.862	0.869	0.883	0.913	0.932	0.746	**0.771**	0.803	**0.813**	0.802	0.822	0.812	0.803	**0.572**	**0.603**	**0.64**	**0.649**	0.639	0.649	0.639	0.633
	wikiSERA-DIS-5	0.875	0.894	0.928	0.919	0.918	0.891	0.93	**0.942**	0.664	0.748	0.782	0.75	0.8	0.787	0.799	**0.844**	0.486	0.565	0.622	0.574	0.631	0.608	0.636	0.681
	wikiSERA-DIS-10	**0.916**	**0.915**	**0.942**	**0.934**	**0.936**	**0.931**	**0.939**	0.933	**0.748**	0.76	**0.808**	0.804	**0.818**	**0.824**	**0.821**	0.831	**0.572**	0.578	0.639	0.636	**0.646**	**0.651**	**0.654**	**0.684**

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.861	0.885	0.906	0.871	0.874	0.858	0.892	0.908	0.716	0.748	0.802	0.801	0.816	0.761	0.747	0.78	0.548	0.579	0.645	0.626	0.64	0.587	0.573	0.614
	SERA-10	0.828	0.904	0.891	0.854	0.869	0.881	0.914	**0.931**	0.776	0.775	**0.824**	0.809	0.799	0.828	0.829	0.818	0.602	0.604	**0.66**	0.641	0.634	0.657	**0.665**	0.648
	SERA-NP-5	0.849	0.888	0.912	0.872	0.852	0.866	0.882	0.893	0.768	0.746	0.776	0.763	0.769	0.812	0.816	0.757	0.599	0.576	0.61	0.604	0.604	0.647	0.659	0.591
	SERA-NP-10	0.836	0.895	0.889	0.858	0.845	0.868	0.892	0.912	**0.78**	0.753	0.815	0.786	0.718	0.801	0.797	0.81	**0.605**	0.59	0.651	0.618	0.546	0.631	0.625	0.648
	SERA-KW-5	0.863	0.886	0.902	0.87	0.872	0.852	0.892	0.904	0.731	0.745	0.787	0.801	0.807	0.745	0.775	0.789	0.564	0.578	0.636	0.629	0.631	0.571	0.601	0.615
	SERA-KW-10	0.827	0.908	0.89	0.854	0.868	0.876	0.916	0.93	0.764	**0.781**	0.822	0.805	0.787	0.821	**0.832**	0.819	0.591	**0.615**	**0.66**	0.641	0.62	0.661	0.661	0.651
	SERA-DIS-5	0.89	0.896	0.931	0.912	0.927	0.896	0.915	0.928	0.697	0.731	0.778	0.764	**0.83**	0.766	0.705	0.802	0.531	0.567	0.612	0.586	**0.664**	0.594	0.539	0.628
	SERA-DIS-10	**0.907**	**0.911**	0.939	0.918	**0.935**	0.924	**0.927**	0.928	0.719	0.736	0.801	0.78	0.811	0.823	0.785	0.825	0.554	0.563	0.64	0.595	0.647	0.657	0.612	0.648
	SERA-DIS-NP-5	0.876	0.897	0.935	0.93	0.916	0.927	0.908	0.921	0.748	0.731	0.777	0.786	0.795	**0.861**	0.787	0.771	0.569	0.566	0.61	0.618	0.625	**0.686**	0.625	0.602
	SERA-DIS-NP-10	0.896	0.911	**0.945**	**0.937**	0.923	**0.937**	0.921	0.928	0.749	0.748	0.813	**0.825**	0.756	0.852	0.789	0.8	0.577	0.592	0.649	**0.656**	0.593	**0.686**	0.624	0.645
	SERA-DIS-KW-5	0.897	0.896	0.929	0.912	0.921	0.89	0.915	0.923	0.71	0.733	0.788	0.787	0.813	0.764	0.749	0.815	0.546	0.571	0.621	0.608	0.644	0.582	0.573	0.636
	SERA-DIS-KW-10	0.906	**0.911**	0.939	0.918	0.93	0.918	**0.927**	0.923	0.722	0.735	0.807	0.786	0.795	0.822	0.788	**0.829**	0.557	0.562	0.64	0.602	0.626	0.657	0.62	**0.658**
	wikiSERA-5	0.834	0.885	0.904	0.877	0.868	0.836	0.887	0.912	0.675	0.755	0.784	0.792	0.79	0.74	0.791	0.816	0.511	0.577	0.625	0.624	0.627	0.565	0.619	0.644
	wikiSERA-10	0.848	0.911	0.885	0.861	0.868	0.882	0.913	0.932	0.748	**0.773**	0.807	**0.814**	0.805	0.824	0.815	0.805	**0.574**	**0.608**	**0.642**	**0.652**	0.642	0.652	0.642	0.635
	wikiSERA-DIS-5	0.876	0.895	0.929	0.919	0.918	0.891	0.929	**0.942**	0.664	0.748	0.785	0.755	0.805	0.789	0.799	**0.846**	0.485	0.567	0.626	0.579	0.636	0.61	0.636	0.683
	wikiSERA-DIS-10	**0.916**	**0.915**	**0.942**	**0.934**	**0.936**	**0.931**	**0.939**	0.933	**0.75**	0.761	**0.811**	0.806	**0.823**	**0.826**	**0.823**	0.834	**0.574**	0.581	0.641	0.639	**0.651**	**0.653**	**0.656**	**0.684**

Table A.22: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_4$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.869	0.908	0.897	0.863	0.845	0.839	0.865	0.873	0.724	0.792	0.812	0.81	0.818	0.806	0.822	0.786	0.538	0.607	0.651	0.631	0.645	0.626	0.654	0.611
	SERA-10	0.807	0.905	0.876	0.845	0.835	0.864	0.866	0.885	0.732	0.806	0.82	0.837	0.817	0.855	0.821	0.825	0.56	**0.632**	0.653	0.667	0.642	0.684	0.654	0.661
	SERA-NP-5	0.879	0.901	0.913	0.862	0.846	0.831	0.854	0.865	0.757	0.767	**0.845**	0.795	0.805	0.812	0.822	0.799	0.582	0.577	**0.683**	0.631	0.63	0.638	0.668	0.626
	SERA-NP-10	0.825	0.903	0.881	0.838	0.844	0.855	0.864	0.884	**0.772**	0.798	0.818	0.784	0.807	0.843	0.84	0.842	0.597	0.616	0.653	0.605	0.631	0.678	0.671	0.667
	SERA-KW-5	0.867	0.902	0.893	0.864	0.84	0.839	0.861	0.871	0.719	0.774	0.813	0.799	0.817	0.8	0.815	0.779	0.541	0.588	0.654	0.625	0.642	0.616	0.641	0.607
	SERA-KW-10	0.805	0.906	0.874	0.846	0.83	0.86	0.864	0.884	0.73	0.803	0.813	0.84	0.812	0.852	0.815	0.831	0.561	0.623	0.643	**0.671**	0.642	0.681	0.648	0.669
	SERA-DIS-5	0.881	0.914	0.938	0.917	0.903	0.897	0.905	0.911	0.687	0.781	0.83	0.827	**0.865**	0.811	0.847	0.804	0.508	0.6	0.665	0.662	**0.702**	0.64	0.688	0.632
	SERA-DIS-10	0.913	**0.932**	0.938	**0.926**	**0.92**	**0.929**	0.923	**0.939**	0.736	**0.811**	0.836	0.837	0.855	**0.864**	0.831	0.851	0.561	0.629	0.669	0.665	0.693	**0.7**	0.662	**0.693**
	SERA-DIS-NP-5	0.9	0.911	0.942	0.914	0.908	0.895	0.901	0.907	0.751	0.77	0.838	0.8	0.813	0.82	0.828	0.829	0.58	0.58	0.678	0.627	0.636	0.653	0.661	0.651
	SERA-DIS-NP-10	**0.917**	0.923	**0.944**	0.92	**0.92**	0.922	**0.928**	0.937	0.763	0.802	0.827	0.791	0.83	0.828	**0.852**	**0.852**	**0.601**	0.612	0.667	0.619	0.662	0.654	**0.692**	0.68
	SERA-DIS-KW-5	0.88	0.91	0.937	0.916	0.898	0.893	0.902	0.908	0.686	0.768	0.828	0.821	0.847	0.779	0.842	0.795	0.503	0.589	0.669	0.657	0.682	0.609	0.682	0.623
	SERA-DIS-KW-10	0.907	0.931	0.936	0.925	0.915	0.926	0.921	0.934	0.74	0.807	0.824	**0.844**	0.854	0.848	0.817	0.841	0.561	0.62	0.653	0.67	0.697	0.684	0.646	0.677
	wikiSERA-5	0.86	0.908	0.903	0.864	0.841	0.832	0.858	0.867	0.685	0.786	**0.841**	0.805	0.789	0.796	0.806	0.782	0.519	0.615	0.684	0.631	0.626	0.605	0.641	0.605
	wikiSERA-10	0.808	0.906	0.868	0.841	0.825	0.852	0.859	0.887	0.741	0.796	0.817	0.841	0.794	0.856	0.807	0.848	0.581	**0.62**	0.656	0.671	0.619	0.684	0.638	0.684
	wikiSERA-DIS-5	0.894	0.915	**0.944**	0.918	0.896	0.895	0.904	0.906	0.701	0.784	0.838	0.822	**0.818**	0.832	**0.846**	0.799	0.525	0.604	**0.688**	0.653	**0.665**	0.646	**0.685**	0.613
	wikiSERA-DIS-10	**0.917**	**0.931**	0.937	**0.923**	**0.911**	**0.925**	**0.921**	**0.939**	**0.786**	**0.799**	0.818	**0.847**	0.809	**0.863**	0.825	**0.856**	**0.604**	0.619	0.65	**0.679**	0.646	**0.698**	0.661	**0.685**

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.869	0.908	0.897	0.863	0.845	0.838	0.865	0.873	0.726	0.794	0.815	0.813	0.821	0.807	0.824	0.788	0.539	0.609	0.654	0.633	0.648	0.628	0.659	0.613
	SERA-10	0.807	0.905	0.876	0.844	0.834	0.864	0.865	0.885	0.734	0.809	0.823	0.838	0.82	0.856	0.823	0.827	0.562	**0.635**	0.655	0.67	0.644	0.686	0.656	0.663
	SERA-NP-5	0.879	0.901	0.913	0.862	0.846	0.83	0.853	0.864	0.759	0.768	**0.848**	0.797	0.809	0.811	0.823	0.801	0.584	0.578	**0.688**	0.633	0.635	0.637	0.667	0.628
	SERA-NP-10	0.825	0.903	0.88	0.837	0.843	0.854	0.863	0.884	**0.775**	0.798	0.822	0.786	0.811	0.844	0.841	0.842	0.6	0.623	0.655	0.608	0.634	0.68	0.671	0.669
	SERA-KW-5	0.866	0.902	0.893	0.863	0.839	0.838	0.861	0.871	0.721	0.776	0.816	0.802	0.821	0.801	0.817	0.782	0.542	0.59	0.657	0.627	0.644	0.618	0.644	0.609
	SERA-KW-10	0.804	0.906	0.874	0.846	0.829	0.86	0.864	0.884	0.732	0.805	0.816	0.841	0.815	0.853	0.817	0.833	0.563	0.625	0.645	**0.674**	0.644	0.684	0.65	0.671
	SERA-DIS-5	0.881	0.915	0.938	0.917	0.903	0.897	0.905	0.911	0.69	0.783	0.832	0.829	**0.868**	0.813	0.849	0.807	0.511	0.602	0.668	0.664	**0.704**	0.643	**0.691**	0.635
	SERA-DIS-10	0.912	**0.932**	0.938	**0.925**	**0.92**	**0.929**	0.923	**0.939**	0.74	**0.813**	0.839	0.839	0.858	**0.865**	0.833	**0.854**	0.563	0.631	0.674	0.667	0.695	**0.702**	0.664	**0.695**
	SERA-DIS-NP-5	0.9	0.911	0.942	0.914	0.907	0.895	0.9	0.907	0.752	0.771	0.84	0.801	0.816	0.821	0.829	0.83	0.581	0.581	0.682	0.629	0.639	0.655	0.66	0.653
	SERA-DIS-NP-10	**0.917**	0.925	**0.944**	0.919	0.919	0.922	**0.928**	0.937	0.766	0.8	0.83	0.793	0.832	0.829	**0.852**	0.853	**0.604**	0.618	0.672	0.621	0.664	0.656	**0.691**	0.682
	SERA-DIS-KW-5	0.88	0.911	0.937	0.916	0.897	0.892	0.902	0.908	0.689	0.771	0.83	0.824	0.852	0.78	0.844	0.797	0.505	0.591	0.672	0.659	0.687	0.612	0.682	0.625
	SERA-DIS-KW-10	0.907	0.931	0.936	**0.925**	0.915	0.926	0.921	0.934	0.743	0.809	0.826	**0.846**	0.857	0.849	0.819	0.843	0.563	0.622	0.655	0.672	0.699	0.686	0.648	0.679
	wikiSERA-5	0.859	0.908	0.903	0.864	0.84	0.832	0.857	0.866	0.686	0.786	**0.844**	0.808	0.792	0.798	0.807	0.785	0.52	0.616	0.688	0.633	0.628	0.607	0.644	0.608
	wikiSERA-10	0.807	0.905	0.868	0.841	0.824	0.851	0.858	0.887	0.743	0.798	0.82	0.843	0.797	0.857	0.809	0.85	0.583	**0.622**	0.658	0.673	0.618	0.686	0.637	0.684
	wikiSERA-DIS-5	0.894	0.915	**0.944**	0.918	0.896	0.894	0.903	0.906	0.704	0.785	0.84	0.824	**0.821**	0.834	**0.848**	0.801	0.526	0.606	**0.691**	0.655	**0.67**	0.648	**0.687**	0.614
	wikiSERA-DIS-10	**0.917**	**0.931**	0.937	**0.923**	**0.91**	**0.924**	**0.921**	**0.94**	**0.789**	**0.8**	0.82	**0.849**	0.811	**0.865**	0.828	**0.859**	**0.605**	0.621	0.655	**0.682**	0.645	**0.701**	0.663	**0.69**

Table A.23: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.884	0.909	0.914	0.879	0.869	0.855	0.881	0.896	0.766	0.795	0.82	0.817	0.856	0.807	0.806	0.805	0.593	0.617	0.661	0.643	0.69	0.631	0.631	0.635
	SERA-10	0.821	0.912	0.891	0.861	0.856	0.874	0.886	0.908	0.768	0.822	0.828	**0.844**	0.813	0.847	0.835	0.826	0.598	0.655	0.662	**0.679**	0.644	0.677	0.664	0.655
	SERA-NP-5	0.886	0.912	0.922	0.873	0.851	0.852	0.873	0.887	0.795	0.796	0.834	0.793	0.812	0.826	0.84	0.812	0.612	0.612	0.675	0.626	0.639	0.658	0.677	0.631
	SERA-NP-10	0.835	0.904	0.893	0.854	0.854	0.864	0.876	0.898	**0.803**	0.798	**0.841**	0.814	0.81	0.841	0.839	0.844	**0.627**	0.623	**0.681**	0.641	0.64	0.674	0.665	0.666
	SERA-KW-5	0.88	0.905	0.908	0.877	0.863	0.851	0.877	0.893	0.778	0.791	0.819	0.813	0.85	0.802	0.813	0.806	0.612	0.619	0.651	0.636	0.685	0.622	0.635	0.633
	SERA-KW-10	0.818	0.913	0.888	0.862	0.851	0.869	0.885	0.906	0.758	**0.828**	0.818	0.838	0.809	0.846	0.833	0.823	0.586	**0.659**	0.654	0.671	0.639	0.675	0.659	0.652
	SERA-DIS-5	0.903	0.917	0.946	0.926	0.922	0.905	0.917	0.925	0.758	0.787	0.836	0.823	**0.874**	0.802	0.802	0.823	0.585	0.608	0.669	0.651	**0.723**	0.632	0.644	0.657
	SERA-DIS-10	**0.92**	**0.932**	0.944	0.933	**0.933**	0.932	**0.931**	**0.941**	0.769	0.816	0.826	0.842	0.853	**0.865**	0.826	0.842	0.607	0.644	0.665	0.671	0.698	**0.702**	0.659	0.674
	SERA-DIS-NP-5	0.905	0.918	0.947	0.926	0.912	0.915	0.91	0.925	0.782	0.785	0.84	0.818	0.817	0.86	0.835	0.836	0.618	0.607	0.678	0.654	0.646	0.693	0.686	0.666
	SERA-DIS-NP-10	0.916	0.928	**0.949**	**0.934**	0.925	**0.933**	**0.931**	**0.941**	0.777	0.815	0.835	0.831	0.833	0.845	**0.853**	**0.858**	0.611	0.642	**0.681**	0.665	0.666	0.686	**0.693**	**0.702**
	SERA-DIS-KW-5	0.902	0.913	0.945	0.924	0.917	0.9	0.914	0.92	0.757	0.783	0.838	0.825	0.865	0.792	0.821	0.805	0.584	0.608	0.671	0.651	0.708	0.622	0.651	0.631
	SERA-DIS-KW-10	0.914	0.931	0.942	0.931	0.928	0.928	0.928	0.936	0.767	0.815	0.822	0.841	0.847	0.851	0.819	0.841	0.597	0.64	0.653	0.663	0.696	0.689	0.647	0.675
	wikiSERA-5	0.87	0.911	0.918	0.881	0.858	0.841	0.872	0.889	0.729	0.803	0.836	0.81	0.807	0.807	0.802	0.814	0.551	0.629	0.679	0.639	0.644	0.608	0.632	0.634
	wikiSERA-10	0.833	0.915	0.885	0.86	0.849	0.864	0.882	0.908	0.758	0.813	0.824	0.844	0.806	0.856	0.826	0.832	0.589	0.642	0.662	0.679	0.639	0.684	0.655	0.663
	wikiSERA-DIS-5	0.903	0.92	**0.951**	0.93	0.911	0.901	0.918	0.927	0.741	0.812	**0.841**	0.817	0.818	0.84	0.833	0.835	0.563	0.638	**0.691**	0.644	0.657	0.657	**0.67**	0.661
	wikiSERA-DIS-10	**0.926**	**0.934**	0.946	**0.937**	**0.927**	**0.931**	**0.934**	**0.944**	**0.794**	**0.823**	0.832	**0.853**	**0.83**	**0.867**	**0.836**	**0.871**	**0.622**	**0.654**	0.677	**0.686**	**0.667**	**0.705**	0.667	**0.704**

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.883	0.909	0.914	0.879	0.869	0.854	0.881	0.896	0.769	0.797	0.824	0.819	0.86	0.807	0.808	0.808	0.595	0.619	0.664	0.645	0.692	0.631	0.633	0.638
	SERA-10	0.821	0.912	0.89	0.861	0.856	0.873	0.886	0.907	0.77	0.825	0.831	**0.845**	0.817	0.849	0.836	0.828	0.6	0.66	0.664	**0.682**	0.646	0.679	0.666	0.657
	SERA-NP-5	0.885	0.912	0.922	0.873	0.85	0.851	0.872	0.887	0.798	0.798	0.837	0.795	0.816	0.826	0.842	0.813	0.614	0.616	0.68	0.629	0.644	0.657	0.679	0.633
	SERA-NP-10	0.834	0.904	0.892	0.854	0.854	0.863	0.876	0.898	**0.807**	0.795	**0.844**	0.816	0.813	0.842	0.84	0.844	**0.629**	0.625	0.683	0.643	0.643	0.676	0.665	0.668
	SERA-KW-5	0.88	0.905	0.908	0.877	0.863	0.85	0.877	0.893	0.781	0.793	0.822	0.816	0.854	0.803	0.814	0.809	0.614	0.621	0.655	0.639	0.687	0.624	0.635	0.635
	SERA-KW-10	0.817	0.913	0.888	0.861	0.85	0.869	0.885	0.906	0.76	**0.831**	0.821	0.84	0.813	0.848	0.835	0.824	0.588	**0.664**	0.656	0.674	0.641	0.678	0.662	0.654
	SERA-DIS-5	0.902	0.917	0.946	0.926	0.922	0.905	0.917	0.925	0.761	0.789	0.838	0.825	**0.879**	0.803	0.804	0.826	0.587	0.61	0.672	0.656	**0.728**	0.635	0.647	0.659
	SERA-DIS-10	**0.919**	**0.932**	0.944	**0.933**	**0.933**	**0.932**	**0.931**	**0.941**	0.772	0.818	0.828	0.844	0.855	**0.867**	0.828	0.844	0.609	0.647	0.667	0.674	0.698	**0.705**	0.662	0.676
	SERA-DIS-NP-5	0.905	0.918	0.947	0.926	0.912	0.915	0.91	0.925	0.785	0.787	0.843	0.82	0.82	0.862	0.836	0.838	0.62	0.61	0.683	0.659	0.651	0.694	0.684	0.668
	SERA-DIS-NP-10	0.916	0.93	**0.949**	**0.933**	0.925	**0.932**	**0.931**	**0.941**	0.781	0.815	0.838	0.833	0.836	0.846	**0.853**	**0.859**	0.613	0.645	**0.686**	0.667	0.668	0.687	**0.692**	**0.702**
	SERA-DIS-KW-5	0.902	0.914	0.945	0.923	0.916	0.899	0.914	0.92	0.76	0.785	0.84	0.828	0.869	0.793	0.823	0.807	0.586	0.61	0.675	0.656	0.713	0.624	0.651	0.633
	SERA-DIS-KW-10	0.913	**0.932**	0.942	0.931	0.928	0.928	0.928	0.936	0.77	0.817	0.824	0.843	0.85	0.852	0.82	0.843	0.6	0.643	0.655	0.666	0.695	0.691	0.649	0.677
	wikiSERA-5	0.87	0.911	0.918	0.881	0.858	0.841	0.871	0.889	0.73	0.803	0.839	0.813	0.81	0.808	0.804	0.816	0.552	0.631	0.683	0.641	0.647	0.61	0.635	0.636
	wikiSERA-10	0.832	0.915	0.885	0.86	0.849	0.863	0.881	0.908	0.76	0.816	0.828	0.845	0.808	0.858	0.828	0.833	0.591	0.644	0.665	0.681	0.639	0.686	0.655	0.663
	wikiSERA-DIS-5	0.903	0.92	**0.951**	0.93	0.911	0.901	0.918	0.927	0.743	0.813	**0.843**	0.821	0.822	0.841	0.835	0.836	0.565	0.64	**0.695**	0.647	0.662	0.659	**0.672**	0.662
	wikiSERA-DIS-10	**0.926**	**0.934**	0.946	**0.937**	**0.926**	**0.931**	**0.934**	**0.944**	**0.797**	**0.824**	0.835	**0.854**	**0.832**	**0.869**	**0.837**	**0.873**	**0.625**	**0.656**	0.679	**0.688**	**0.667**	**0.707**	0.67	**0.706**

Table A.24: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.871	0.896	0.901	0.863	0.857	0.842	0.877	0.89	0.712	0.746	0.786	0.797	0.837	0.767	0.794	0.797	0.548	0.57	0.617	0.626	0.674	0.585	0.625	0.621
	SERA-10	0.825	0.907	0.877	0.845	0.852	0.872	0.891	0.91	0.774	**0.801**	0.801	0.809	0.834	**0.843**	0.816	0.833	0.6	**0.633**	0.638	0.636	0.674	**0.677**	0.647	0.665
	SERA-NP-5	0.867	0.888	0.912	0.868	0.855	0.852	0.867	0.884	0.745	0.741	**0.815**	0.783	0.813	0.814	**0.823**	0.809	0.573	0.56	**0.658**	0.618	0.639	0.641	0.657	0.633
	SERA-NP-10	0.838	0.899	0.881	0.849	0.842	0.863	0.88	0.899	**0.793**	0.786	0.807	0.779	0.771	0.84	0.821	0.828	**0.62**	0.622	0.646	0.601	0.597	0.676	0.655	0.658
	SERA-KW-5	0.871	0.893	0.898	0.864	0.854	0.84	0.873	0.889	0.724	0.744	0.78	0.796	0.826	0.755	0.797	0.792	0.559	0.562	0.614	0.622	0.659	0.576	0.624	0.618
	SERA-KW-10	0.823	0.909	0.877	0.847	0.849	0.868	0.89	0.91	0.756	0.8	0.8	0.812	0.822	0.841	0.818	0.834	0.582	0.628	0.636	0.64	0.657	0.675	0.654	0.666
	SERA-DIS-5	0.883	0.902	0.937	0.917	0.914	0.894	0.915	0.921	0.654	0.723	0.807	0.8	**0.858**	0.765	0.801	0.804	0.49	0.546	0.639	0.626	**0.708**	0.585	0.631	0.628
	SERA-DIS-10	**0.914**	**0.923**	0.935	0.924	**0.93**	**0.926**	**0.929**	**0.938**	0.731	0.779	0.803	0.822	0.851	0.834	0.81	0.843	0.57	0.592	0.636	**0.653**	0.692	0.667	0.65	0.685
	SERA-DIS-NP-5	0.894	0.897	0.941	0.922	0.918	0.908	0.906	0.915	0.704	0.728	0.812	0.788	0.824	0.83	0.795	0.811	0.534	0.541	0.65	0.622	0.648	0.658	0.631	0.639
	SERA-DIS-NP-10	0.913	0.918	**0.944**	**0.929**	0.922	**0.926**	0.927	0.93	0.74	0.781	0.805	0.803	0.815	0.832	0.822	0.825	0.572	0.613	0.643	0.632	0.644	0.653	**0.659**	0.663
	SERA-DIS-KW-5	0.887	0.9	0.936	0.917	0.909	0.89	0.912	0.918	0.674	0.714	0.81	0.809	0.846	0.744	0.797	0.793	0.504	0.537	0.646	0.634	0.689	0.564	0.628	0.618
	SERA-DIS-KW-10	0.91	**0.923**	0.935	0.924	0.925	0.921	0.928	0.933	0.726	0.778	0.799	**0.825**	0.845	0.816	0.808	**0.852**	0.566	0.597	0.627	**0.653**	0.684	0.643	0.649	**0.689**
	wikiSERA-5	0.854	0.895	0.904	0.87	0.856	0.83	0.869	0.887	0.664	0.767	**0.819**	0.797	0.802	0.739	0.792	0.803	0.499	0.587	**0.653**	0.622	0.639	0.556	0.616	0.628
	wikiSERA-10	0.829	0.907	0.868	0.849	0.844	0.865	0.884	0.914	**0.779**	**0.791**	0.791	0.823	0.802	**0.84**	0.8	0.833	**0.599**	**0.624**	0.63	**0.657**	0.635	**0.674**	0.631	0.674
	wikiSERA-DIS-5	0.886	0.903	**0.94**	0.919	0.908	0.892	0.917	0.922	0.662	0.758	0.807	0.785	0.823	0.78	0.81	0.801	0.492	0.574	0.65	0.612	0.662	0.605	0.634	0.622
	wikiSERA-DIS-10	**0.918**	**0.923**	0.934	**0.927**	**0.924**	**0.926**	**0.931**	**0.94**	0.775	0.781	0.794	**0.828**	**0.827**	0.828	**0.819**	**0.843**	0.596	0.6	0.635	**0.657**	**0.665**	0.659	**0.654**	**0.681**

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.871	0.896	0.901	0.863	0.857	0.842	0.877	0.89	0.714	0.748	0.79	0.801	0.841	0.768	0.796	0.8	0.549	0.572	0.622	0.628	0.676	0.587	0.627	0.626
	SERA-10	0.825	0.907	0.876	0.845	0.851	0.871	0.89	0.91	0.776	**0.803**	0.804	0.81	0.838	**0.845**	0.818	0.835	0.602	**0.635**	0.64	0.639	0.676	**0.679**	0.649	0.667
	SERA-NP-5	0.867	0.888	0.911	0.868	0.855	0.851	0.866	0.883	0.747	0.743	**0.818**	0.785	0.817	0.814	**0.824**	0.811	0.575	0.56	**0.663**	0.62	0.644	0.641	0.656	0.635
	SERA-NP-10	0.838	0.897	0.881	0.849	0.842	0.863	0.88	0.899	**0.796**	0.754	0.81	0.78	0.774	0.841	0.822	0.829	**0.622**	0.622	0.648	0.601	0.6	0.675	0.655	0.657
	SERA-KW-5	0.871	0.893	0.897	0.864	0.854	0.84	0.872	0.888	0.726	0.746	0.783	0.8	0.83	0.756	0.799	0.796	0.56	0.564	0.619	0.625	0.661	0.578	0.626	0.621
	SERA-KW-10	0.822	0.909	0.876	0.846	0.848	0.868	0.89	0.91	0.758	**0.803**	0.803	0.814	0.825	0.842	0.821	0.836	0.584	0.631	0.639	0.643	0.659	0.678	0.656	0.668
	SERA-DIS-5	0.883	0.902	0.937	0.917	0.913	0.894	0.915	0.921	0.657	0.725	0.81	0.803	**0.862**	0.766	0.802	0.807	0.492	0.548	0.643	0.628	**0.713**	0.587	0.633	0.631
	SERA-DIS-10	**0.914**	**0.923**	0.935	0.924	**0.93**	**0.926**	**0.929**	**0.938**	0.734	0.781	0.806	0.824	0.854	0.835	0.813	0.846	0.573	0.594	0.639	**0.655**	0.694	0.669	0.652	0.687
	SERA-DIS-NP-5	0.894	0.897	0.941	0.922	0.918	0.908	0.906	0.915	0.705	0.73	0.813	0.789	0.828	0.83	0.795	0.813	0.536	0.543	0.652	0.626	0.653	0.66	0.631	0.639
	SERA-DIS-NP-10	0.913	0.918	**0.944**	**0.928**	0.922	**0.926**	0.927	0.931	0.743	0.784	0.807	0.805	0.817	0.833	0.823	0.826	0.574	0.602	0.645	0.635	0.646	0.655	**0.659**	0.663
	SERA-DIS-KW-5	0.887	0.9	0.936	0.917	0.908	0.889	0.912	0.918	0.676	0.716	0.812	0.812	0.851	0.745	0.799	0.796	0.507	0.539	0.649	0.636	0.694	0.566	0.631	0.62
	SERA-DIS-KW-10	0.91	**0.923**	0.935	0.924	0.925	0.921	0.928	0.933	0.728	0.781	0.801	**0.828**	0.848	0.817	0.811	**0.854**	0.569	0.6	0.629	**0.655**	0.686	0.645	0.652	**0.691**
	wikiSERA-5	0.853	0.895	0.904	0.87	0.855	0.83	0.869	0.887	0.665	0.767	**0.823**	0.8	0.806	0.741	0.793	0.806	0.5	0.59	**0.657**	0.624	0.641	0.558	0.618	0.63
	wikiSERA-10	0.828	0.907	0.868	0.849	0.844	0.865	0.884	0.913	**0.782**	**0.794**	0.795	0.825	0.805	**0.842**	0.801	0.835	**0.601**	**0.626**	0.632	**0.659**	0.635	**0.676**	0.633	0.673
	wikiSERA-DIS-5	0.886	0.903	**0.94**	0.919	0.908	0.892	0.916	0.921	0.664	0.759	0.809	0.789	0.828	0.782	0.812	0.804	0.494	0.577	0.653	0.614	**0.666**	0.608	0.636	0.624
	wikiSERA-DIS-10	**0.918**	**0.923**	0.934	**0.926**	**0.924**	**0.926**	**0.931**	**0.94**	0.778	0.783	0.797	**0.83**	**0.83**	0.83	**0.821**	**0.846**	0.598	0.602	0.637	**0.659**	0.664	0.662	**0.656**	**0.683**

Table A.25: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.874	0.907	0.904	0.867	0.855	0.847	0.877	0.889	0.725	0.794	0.818	0.808	0.827	0.803	0.817	0.798	0.544	0.612	0.656	0.637	0.664	0.622	0.654	0.623
	SERA-10	0.815	0.908	0.881	0.848	0.846	0.87	0.882	0.903	0.755	0.802	0.817	0.83	0.821	0.853	0.825	0.83	0.584	0.628	0.651	0.663	0.652	0.684	0.662	0.665
	SERA-NP-5	0.877	0.903	0.916	0.868	0.851	0.843	0.867	0.881	0.768	0.768	0.831	0.797	0.802	0.816	0.832	0.809	0.59	0.582	0.67	0.635	0.633	0.652	0.675	0.635
	SERA-NP-10	0.83	0.902	0.885	0.845	0.846	0.861	0.874	0.896	**0.78**	0.793	0.828	0.79	0.788	0.843	0.833	0.845	**0.6**	0.624	0.662	0.615	0.617	0.679	0.67	0.681
	SERA-KW-5	0.873	0.903	0.9	0.867	0.85	0.845	0.873	0.887	0.741	0.788	0.816	0.804	0.818	0.793	0.826	0.796	0.562	0.609	0.654	0.631	0.651	0.613	0.66	0.628
	SERA-KW-10	0.812	0.91	0.88	0.85	0.842	0.866	0.882	0.903	0.743	**0.805**	0.81	0.832	0.81	0.852	0.819	0.831	0.573	**0.635**	0.646	0.663	0.646	0.688	0.657	0.667
	SERA-DIS-5	0.89	0.914	0.942	0.92	0.913	0.901	0.916	0.923	0.71	0.778	0.829	0.813	**0.865**	0.798	0.833	0.82	0.528	0.597	0.666	0.635	**0.71**	0.62	0.675	0.653
	SERA-DIS-10	0.914	**0.929**	0.941	0.926	**0.928**	**0.93**	**0.93**	**0.942**	0.742	0.795	**0.836**	0.829	0.853	**0.857**	0.83	0.853	0.576	0.619	**0.671**	0.657	0.696	**0.693**	0.669	0.69
	SERA-DIS-NP-5	0.901	0.912	0.945	0.922	0.913	0.907	0.908	0.92	0.753	0.77	0.831	0.812	0.818	0.839	0.835	0.828	0.587	0.59	0.67	0.65	0.65	0.668	0.675	0.648
	SERA-DIS-NP-10	**0.916**	0.927	**0.947**	**0.927**	0.922	0.928	**0.93**	0.939	0.766	0.794	0.825	0.819	0.827	0.84	**0.851**	0.847	0.599	0.619	0.665	0.65	0.659	0.675	**0.69**	0.685
	SERA-DIS-KW-5	0.891	0.912	0.941	0.919	0.908	0.896	0.914	0.92	0.691	0.768	0.822	0.819	0.854	0.788	0.838	0.808	0.51	0.591	0.659	0.644	0.694	0.62	0.679	0.635
	SERA-DIS-KW-10	0.91	**0.929**	0.939	0.926	0.923	0.926	0.929	0.937	0.744	0.79	0.826	**0.841**	0.851	0.84	0.821	**0.854**	0.565	0.616	0.658	**0.666**	0.693	0.674	0.658	**0.694**
	wikiSERA-5	0.86	0.907	0.908	0.869	0.85	0.835	0.869	0.884	0.695	0.792	0.835	0.8	0.786	0.79	0.814	0.799	0.519	0.619	0.678	0.63	0.626	0.6	0.65	0.624
	wikiSERA-10	0.821	0.91	0.874	0.848	0.838	0.862	0.877	0.905	0.748	**0.805**	0.813	0.836	0.8	0.85	0.817	0.842	0.578	**0.637**	0.652	**0.673**	0.642	0.677	0.648	0.682
	wikiSERA-DIS-5	0.894	0.915	**0.947**	0.922	0.906	0.897	0.917	0.923	0.709	0.787	**0.836**	0.809	0.817	0.822	**0.839**	0.815	0.533	0.612	**0.688**	0.641	0.659	0.64	**0.675**	0.639
	wikiSERA-DIS-10	**0.921**	**0.93**	0.94	**0.929**	**0.921**	**0.929**	**0.932**	**0.944**	**0.783**	0.797	0.818	**0.844**	**0.832**	**0.864**	0.837	**0.861**	**0.603**	0.622	0.657	**0.673**	**0.671**	**0.701**	0.672	**0.693**

	Method	Pearson								Spearman								Kendall							
		TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
			825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.874	0.907	0.904	0.867	0.854	0.847	0.876	0.888	0.728	0.796	0.821	0.811	0.831	0.803	0.819	0.8	0.545	0.614	0.659	0.639	0.666	0.624	0.656	0.626
	SERA-10	0.814	0.908	0.881	0.848	0.845	0.87	0.882	0.903	0.757	0.804	0.821	0.832	0.824	0.855	0.827	0.832	0.586	0.63	0.653	0.666	0.654	0.686	0.664	0.667
	SERA-NP-5	0.877	0.903	0.916	0.867	0.85	0.842	0.866	0.881	0.771	0.77	0.834	0.799	0.806	0.816	0.833	0.811	0.593	0.582	**0.675**	0.638	0.638	0.651	0.675	0.637
	SERA-NP-10	0.83	0.904	0.884	0.844	0.845	0.86	0.874	0.896	**0.784**	0.796	0.832	0.792	0.792	0.844	0.835	0.846	**0.602**	0.615	0.665	0.617	0.622	0.682	0.67	0.684
	SERA-KW-5	0.873	0.904	0.9	0.867	0.85	0.845	0.873	0.886	0.743	0.789	0.819	0.807	0.822	0.793	0.827	0.799	0.562	0.611	0.657	0.633	0.653	0.615	0.66	0.63
	SERA-KW-10	0.812	0.91	0.88	0.849	0.842	0.866	0.882	0.903	0.745	**0.808**	0.814	0.834	0.814	0.853	0.822	0.833	0.576	**0.637**	0.648	0.665	0.648	0.69	0.659	0.669
	SERA-DIS-5	0.89	0.914	0.942	0.919	0.912	0.901	0.916	0.923	0.712	0.78	0.831	0.816	**0.87**	0.8	0.835	0.823	0.53	0.6	0.67	0.637	**0.715**	0.622	0.678	0.655
	SERA-DIS-10	0.914	**0.929**	0.941	0.926	**0.928**	**0.93**	**0.93**	**0.942**	0.745	0.797	**0.838**	0.831	0.856	**0.858**	0.833	0.855	0.578	0.621	0.674	0.659	0.695	**0.695**	0.671	0.695
	SERA-DIS-NP-5	0.901	0.912	0.945	0.922	0.913	0.906	0.908	0.92	0.755	0.772	0.833	0.814	0.822	0.84	0.836	0.829	0.589	0.591	**0.675**	0.655	0.652	0.67	0.675	0.651
	SERA-DIS-NP-10	**0.916**	0.927	**0.947**	**0.927**	0.922	0.928	**0.93**	0.939	0.77	0.795	0.827	0.821	0.83	0.84	**0.851**	0.849	0.601	0.62	0.67	0.652	0.662	0.676	**0.689**	0.684
	SERA-DIS-KW-5	0.891	0.912	0.941	0.919	0.907	0.896	0.914	0.92	0.693	0.77	0.824	0.822	0.858	0.789	0.839	0.81	0.512	0.593	0.663	0.647	0.699	0.622	0.679	0.637
	SERA-DIS-KW-10	0.91	**0.929**	0.939	0.926	0.923	0.926	0.929	0.938	0.747	0.792	0.829	**0.842**	0.854	0.841	0.823	**0.856**	0.567	0.618	0.66	**0.668**	0.695	0.676	0.66	**0.697**
	wikiSERA-5	0.86	0.907	0.908	0.869	0.85	0.835	0.869	0.884	0.697	0.792	**0.838**	0.803	0.789	0.791	0.816	0.801	0.52	0.62	0.681	0.632	0.628	0.602	0.652	0.626
	wikiSERA-10	0.82	0.91	0.874	0.848	0.838	0.861	0.876	0.905	0.751	**0.807**	0.816	0.838	0.803	0.852	0.818	0.844	0.581	**0.639**	0.654	**0.675**	0.641	0.679	0.651	0.682
	wikiSERA-DIS-5	0.895	0.915	**0.947**	0.922	0.905	0.897	0.917	0.923	0.711	0.787	**0.838**	0.813	0.82	0.824	**0.841**	0.818	0.534	0.614	**0.691**	0.644	0.664	0.643	**0.678**	0.64
	wikiSERA-DIS-10	**0.92**	**0.931**	0.94	**0.929**	**0.921**	**0.928**	**0.932**	**0.944**	**0.786**	0.798	0.821	**0.845**	**0.834**	**0.866**	0.84	**0.864**	**0.605**	0.624	0.659	**0.675**	**0.67**	**0.703**	0.675	**0.698**

Table A.26: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.4 Correlation of SERA and wikiSERA with Responsiveness on TAC2009/AQUAINT-2

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.792	0.807	0.821	0.817	0.802	0.796	0.823	0.816	0.637	0.699	0.685	0.697	0.706	0.685	0.751	0.698	0.474	0.526	0.524	0.547	0.545	0.508	0.572	0.522
SERA-10	0.781	**0.833**	0.83	0.805	0.785	0.814	0.804	0.813	0.697	0.699	**0.729**	0.713	0.679	0.746	0.698	0.698	0.507	0.53	**0.571**	0.546	0.517	0.559	0.532	0.539
SERA-NP-5	0.795	0.808	**0.837**	0.805	0.778	0.754	0.821	0.851	0.666	0.676	0.718	0.664	0.655	0.636	0.743	0.738	0.5	0.506	0.548	0.512	0.499	0.475	0.58	0.59
SERA-NP-10	0.784	0.82	0.813	0.777	0.807	0.823	0.827	0.844	**0.727**	0.689	0.707	0.669	0.718	0.73	0.744	0.755	0.55	0.506	0.533	0.513	0.554	0.56	0.582	0.584
SERA-KW-5	0.779	0.795	0.813	0.809	0.791	0.789	0.821	0.803	0.625	0.684	0.68	0.681	0.708	0.673	0.757	0.699	0.46	0.505	0.52	0.526	0.545	0.498	0.575	0.526
SERA-KW-10	0.781	0.832	0.819	0.801	0.778	0.807	0.801	0.803	0.71	0.697	0.717	**0.72**	0.679	0.743	0.701	0.677	0.518	0.531	0.559	**0.548**	0.516	0.556	0.531	0.518
SERA-DIS-5	0.762	0.784	0.819	**0.826**	0.822	0.817	0.829	0.827	0.617	0.69	0.722	0.692	**0.73**	0.698	0.738	0.739	0.451	0.513	0.54	0.521	**0.581**	0.526	0.548	0.556
SERA-DIS-10	**0.808**	0.803	0.825	0.822	**0.826**	0.828	0.821	0.834	0.681	0.677	0.721	0.703	0.725	**0.758**	0.74	0.753	0.51	0.511	0.552	0.531	0.561	**0.577**	0.553	0.58
SERA-DIS-NP-5	0.757	0.781	0.831	0.822	0.796	0.813	0.822	0.841	0.603	**0.717**	0.694	0.711	0.634	0.674	0.756	0.756	0.453	**0.544**	0.529	0.537	0.479	0.51	0.569	0.591
SERA-DIS-NP-10	0.804	0.807	0.814	0.804	0.814	**0.845**	**0.836**	**0.855**	0.724	0.684	0.722	0.669	0.7	0.727	**0.77**	**0.78**	**0.557**	0.519	0.553	0.505	0.549	0.554	**0.602**	**0.599**
SERA-DIS-KW-5	0.753	0.776	0.814	0.813	0.81	0.807	0.827	0.807	0.602	0.679	0.713	0.681	0.719	0.679	0.754	0.723	0.441	0.499	0.549	0.509	0.563	0.508	0.568	0.537
SERA-DIS-KW-10	0.804	0.803	0.814	0.81	0.815	0.826	0.815	0.819	0.683	0.693	0.706	0.693	0.709	0.745	0.734	0.726	0.502	0.525	0.534	0.521	0.546	0.561	0.55	0.552
wikiSERA-5	0.781	0.8	**0.849**	0.823	0.786	0.798	0.814	0.82	0.644	0.688	0.757	0.713	0.658	0.699	0.711	0.726	0.475	0.52	0.587	0.54	0.495	0.533	0.55	0.545
wikiSERA-10	0.783	**0.83**	0.835	0.806	0.78	0.797	0.803	0.825	0.683	**0.693**	0.742	0.732	0.661	**0.745**	0.68	0.718	0.499	**0.521**	0.573	0.563	0.512	0.557	0.506	0.558
wikiSERA-DIS-5	0.756	0.782	0.841	**0.849**	**0.815**	0.822	0.825	0.831	0.665	0.675	0.749	0.747	0.692	0.716	**0.748**	0.743	0.505	0.513	0.579	0.575	0.526	0.541	**0.565**	0.56
wikiSERA-DIS-10	**0.809**	0.8	0.842	0.844	**0.815**	**0.831**	**0.827**	**0.836**	**0.739**	0.691	**0.762**	**0.765**	**0.712**	0.743	0.74	**0.747**	**0.577**	0.519	**0.588**	**0.59**	**0.559**	**0.565**	0.553	**0.563**

Table A.27: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.813	0.786	0.804	0.818	0.818	0.771	**0.808**	0.793	0.66	0.638	0.626	0.685	0.794	0.614	0.679	0.658	0.501	0.476	0.479	0.516	0.601	0.454	0.512	0.491
SERA-10	**0.839**	**0.823**	0.81	0.794	0.798	0.809	0.803	0.819	**0.777**	**0.684**	0.668	**0.7**	0.726	0.723	0.678	0.706	**0.601**	**0.524**	0.509	0.518	0.548	0.547	0.516	0.539
SERA-NP-5	0.815	0.761	**0.82**	0.784	0.788	0.793	0.795	0.796	0.659	0.622	**0.699**	0.625	0.704	0.722	**0.705**	0.685	0.493	0.47	0.527	0.465	0.536	0.548	**0.543**	0.512
SERA-NP-10	0.818	0.814	0.801	0.781	0.792	0.797	0.804	**0.833**	0.757	0.653	0.676	0.669	0.705	**0.742**	0.68	**0.765**	0.582	0.505	0.499	0.495	0.543	**0.552**	0.511	**0.583**
SERA-KW-5	0.803	0.779	0.796	0.811	0.809	0.763	0.79	0.785	0.65	0.645	0.616	0.678	**0.797**	0.606	0.659	0.638	0.487	0.477	0.47	0.503	**0.607**	0.448	0.503	0.473
SERA-KW-10	0.828	0.822	0.804	0.791	0.788	0.8	0.796	0.809	0.754	0.669	0.656	0.699	0.709	0.706	0.666	0.711	0.584	0.515	0.496	0.52	0.529	0.533	0.505	0.545
SERA-DIS-5	0.769	0.756	0.797	**0.829**	**0.823**	0.792	0.788	0.765	0.634	0.636	0.625	0.697	0.753	0.595	0.646	0.628	0.483	0.472	0.472	**0.528**	0.592	0.444	0.488	0.478
SERA-DIS-10	0.819	0.788	0.798	0.81	0.818	**0.81**	0.779	0.781	0.738	0.681	0.654	0.69	0.729	0.716	0.654	0.719	0.577	0.514	0.499	0.51	0.559	0.544	0.486	0.546
SERA-DIS-NP-5	0.783	0.713	0.804	0.779	0.805	0.802	0.778	0.76	0.644	0.594	0.692	0.602	0.717	0.68	0.613	0.596	0.488	0.441	**0.53**	0.45	0.546	0.515	0.457	0.442
SERA-DIS-NP-10	0.808	0.802	0.798	0.788	0.793	0.789	0.789	0.783	0.737	0.662	0.671	0.652	0.695	0.676	0.652	0.694	0.572	0.501	0.51	0.48	0.534	0.499	0.495	0.514
SERA-DIS-KW-5	0.767	0.746	0.794	0.819	0.812	0.782	0.768	0.751	0.65	0.615	0.618	0.683	0.74	0.579	0.611	0.609	0.485	0.456	0.457	0.514	0.571	0.43	0.458	0.464
SERA-DIS-KW-10	0.805	0.786	0.795	0.802	0.806	0.8	0.768	0.764	0.711	0.668	0.637	0.677	0.732	0.694	0.624	0.685	0.548	0.506	0.487	0.499	0.554	0.519	0.459	0.519
wikiSERA-5	0.801	0.777	**0.822**	**0.816**	0.815	0.772	0.79	0.811	0.648	0.62	0.658	0.683	**0.75**	0.615	0.632	0.696	0.472	0.46	**0.5**	0.512	**0.593**	0.455	0.482	0.527
wikiSERA-10	**0.821**	**0.818**	0.809	0.793	0.783	0.799	**0.801**	**0.825**	**0.754**	**0.66**	**0.663**	**0.708**	0.688	**0.717**	**0.673**	**0.747**	**0.587**	**0.51**	0.494	**0.532**	0.516	**0.532**	**0.499**	0.567
wikiSERA-DIS-5	0.778	0.758	0.818	0.809	**0.825**	0.796	0.786	0.8	0.62	0.638	0.635	0.661	0.717	0.628	0.61	0.681	0.449	0.478	0.478	0.495	0.546	0.472	0.464	0.513
wikiSERA-DIS-10	0.792	0.78	0.806	0.801	0.812	**0.805**	0.785	0.795	0.695	0.653	0.653	0.674	0.711	0.672	0.63	0.741	0.522	0.509	0.488	0.503	0.542	0.511	0.472	**0.568**

Table A.28: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.688	0.706	0.771	0.772	0.745	0.752	0.777	0.773	0.36	0.432	0.584	0.639	0.582	0.615	0.597	0.613	0.264	0.33	0.422	**0.473**	0.44	0.442	0.437	0.455
SERA-10	**0.782**	**0.774**	0.779	0.748	0.762	**0.787**	0.802	0.812	0.642	0.518	0.631	0.626	0.642	0.634	0.662	0.677	0.493	0.389	0.472	0.466	0.479	0.473	0.498	0.511
SERA-NP-5	0.758	0.696	0.777	0.771	0.779	0.746	0.792	0.813	0.498	0.469	0.622	0.608	0.628	0.628	**0.689**	0.674	0.381	0.355	0.477	0.454	0.479	0.473	**0.538**	0.515
SERA-NP-10	0.769	0.725	0.778	0.749	0.753	0.786	**0.806**	**0.814**	0.618	0.483	0.636	0.611	0.583	**0.651**	0.683	0.644	0.454	0.334	0.476	0.447	0.422	**0.489**	0.512	0.502
SERA-KW-5	0.698	0.696	0.769	0.773	0.738	0.751	0.775	0.77	0.372	0.419	0.602	**0.641**	0.577	0.611	0.598	0.619	0.266	0.315	0.446	**0.473**	0.432	0.438	0.444	0.458
SERA-KW-10	0.779	**0.774**	0.776	0.748	0.759	0.783	0.803	0.81	**0.646**	**0.523**	0.616	0.639	**0.654**	0.632	0.663	0.683	**0.497**	**0.39**	0.46	0.472	**0.491**	0.472	0.505	0.527
SERA-DIS-5	0.649	0.686	0.767	0.776	0.775	0.758	0.792	0.775	0.285	0.412	0.585	0.617	0.611	0.578	0.662	0.615	0.2	0.302	0.434	0.455	0.455	0.422	0.475	0.464
SERA-DIS-10	0.761	0.738	0.779	0.771	**0.787**	0.784	0.796	0.801	0.491	0.495	0.616	0.619	0.642	0.607	0.646	0.699	0.356	0.372	0.456	0.46	0.471	0.453	0.478	0.53
SERA-DIS-NP-5	0.74	0.685	0.78	**0.788**	0.786	0.755	0.792	0.788	0.41	0.476	0.617	0.559	0.644	0.594	0.67	0.656	0.293	0.359	0.468	0.42	0.479	0.437	0.488	0.496
SERA-DIS-NP-10	0.778	0.763	**0.79**	0.777	0.77	0.781	0.796	0.789	0.531	0.503	**0.658**	0.598	0.588	0.625	0.683	0.656	0.389	0.38	**0.498**	0.444	0.437	0.464	0.501	0.509
SERA-DIS-KW-5	0.661	0.685	0.768	0.774	0.762	0.748	0.793	0.774	0.303	0.412	0.584	0.604	0.587	0.563	0.649	0.608	0.215	0.299	0.437	0.441	0.436	0.409	0.465	0.453
SERA-DIS-KW-10	0.763	0.742	0.779	0.77	0.777	0.778	0.8	0.793	0.523	0.499	0.631	0.615	0.643	0.581	0.652	**0.703**	0.383	0.378	0.474	0.455	0.467	0.43	0.477	**0.537**
wikiSERA-5	0.698	0.696	**0.799**	0.782	0.747	0.74	0.797	0.771	0.357	0.373	**0.678**	0.659	0.561	0.55	0.667	0.588	0.253	0.283	**0.513**	**0.491**	0.421	0.411	**0.512**	0.44
wikiSERA-10	**0.763**	**0.773**	0.777	0.758	0.748	0.791	0.787	**0.82**	**0.605**	**0.541**	0.639	**0.664**	**0.627**	0.644	0.648	**0.686**	**0.454**	**0.409**	0.473	0.489	**0.475**	0.488	0.479	**0.53**
wikiSERA-DIS-5	0.686	0.685	0.794	**0.793**	**0.778**	0.778	**0.808**	0.776	0.364	0.383	0.615	0.614	0.607	0.568	**0.691**	0.581	0.248	0.29	0.455	0.452	0.448	0.416	0.505	0.434
wikiSERA-DIS-10	0.754	0.736	0.786	0.784	**0.778**	**0.801**	0.794	0.804	0.478	0.502	0.6	0.603	0.606	**0.672**	0.672	0.668	0.349	0.381	0.444	0.444	0.442	**0.509**	0.502	0.507

Table A.29: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_3$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.805	0.753	0.806	**0.804**	0.819	0.772	0.793	0.797	0.593	0.576	0.633	0.686	**0.722**	0.602	0.634	0.646	0.444	0.428	0.479	**0.516**	0.542	0.445	0.473	0.484
SERA-10	**0.817**	0.817	0.812	0.771	0.805	0.799	0.807	0.804	**0.766**	0.625	0.669	0.646	0.698	0.712	0.71	0.691	**0.587**	0.473	0.507	0.486	0.532	0.519	0.537	0.533
SERA-NP-5	0.791	0.736	0.788	0.782	0.796	0.792	**0.817**	**0.821**	0.693	0.593	0.607	0.63	0.661	0.679	**0.75**	0.7	0.52	0.445	0.459	0.475	0.496	0.52	**0.592**	0.534
SERA-NP-10	0.806	0.78	**0.815**	0.782	0.774	0.801	0.808	0.81	0.729	0.605	**0.706**	0.635	0.579	0.699	0.71	0.692	0.546	0.443	**0.532**	0.469	0.438	0.525	0.533	0.526
SERA-KW-5	0.81	0.753	0.795	0.796	0.816	0.764	0.797	0.791	0.604	0.578	0.613	**0.687**	0.721	0.583	0.677	0.686	0.454	0.432	0.466	0.512	0.539	0.432	0.512	0.503
SERA-KW-10	0.812	**0.818**	0.806	0.77	0.8	0.794	0.811	0.794	0.742	**0.631**	0.669	0.647	0.691	0.709	0.715	0.696	0.568	**0.483**	0.506	0.491	0.525	0.525	0.538	0.531
SERA-DIS-5	0.787	0.734	0.8	0.793	**0.827**	0.77	0.765	0.778	0.548	0.582	0.623	0.644	0.718	0.586	0.549	0.668	0.403	0.448	0.469	0.479	**0.554**	0.44	0.393	0.499
SERA-DIS-10	0.809	0.754	0.797	0.765	0.813	0.782	0.757	0.755	0.689	0.586	0.656	0.594	0.698	0.652	0.637	**0.721**	0.519	0.442	0.498	0.438	0.532	0.488	0.472	**0.536**
SERA-DIS-NP-5	0.758	0.708	0.778	0.794	0.82	**0.807**	0.805	0.809	0.616	0.584	0.609	0.65	0.684	0.709	0.689	0.681	0.459	0.432	0.455	0.493	0.519	0.523	0.53	0.517
SERA-DIS-NP-10	0.773	0.787	0.8	0.791	0.796	0.804	0.778	0.777	0.667	0.61	0.681	0.669	0.646	**0.729**	0.671	0.664	0.494	0.447	0.518	0.506	0.506	**0.545**	0.498	0.503
SERA-DIS-KW-5	0.795	0.733	0.794	0.784	0.818	0.76	0.764	0.773	0.557	0.58	0.629	0.655	0.71	0.579	0.594	0.7	0.414	0.441	0.472	0.49	0.542	0.425	0.434	0.515
SERA-DIS-KW-10	0.805	0.751	0.795	0.762	0.803	0.773	0.754	0.744	0.68	0.584	0.658	0.598	0.686	0.654	0.639	0.718	0.509	0.436	0.492	0.442	0.525	0.488	0.475	0.534
wikiSERA-5	0.758	0.763	0.816	**0.81**	0.818	0.764	0.807	**0.816**	0.543	0.611	0.622	**0.671**	0.729	0.593	0.702	0.719	0.403	0.462	0.473	**0.505**	**0.556**	0.439	0.537	0.55
wikiSERA-10	**0.821**	**0.812**	**0.817**	0.777	0.809	**0.807**	**0.814**	0.809	**0.716**	0.617	0.675	0.644	0.689	**0.728**	**0.718**	0.691	**0.547**	**0.471**	0.508	0.489	0.526	**0.542**	**0.542**	0.531
wikiSERA-DIS-5	0.753	0.733	0.81	0.806	**0.821**	0.777	0.801	0.804	0.515	**0.621**	0.643	0.651	**0.73**	0.625	0.668	**0.721**	0.375	0.463	0.492	0.485	0.554	0.459	0.513	**0.552**
wikiSERA-DIS-10	0.801	0.749	0.812	0.786	0.817	0.798	0.781	0.761	0.678	0.603	**0.676**	0.625	0.715	0.682	0.694	0.719	0.499	0.463	**0.515**	0.469	0.549	0.509	0.519	**0.552**

Table A.30: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_4$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.788	0.786	0.81	0.812	0.795	0.781	0.813	0.803	0.596	0.632	0.655	0.696	0.693	0.659	0.699	0.658	0.424	0.472	0.5	0.515	0.522	0.482	0.525	0.486
SERA-10	0.808	**0.819**	0.811	0.787	0.785	0.808	0.81	0.823	0.722	**0.663**	0.683	0.693	0.689	0.714	0.686	0.693	0.54	**0.506**	0.518	0.518	0.511	0.534	0.518	0.53
SERA-NP-5	**0.814**	0.769	**0.822**	0.794	0.788	0.773	0.812	0.83	0.649	0.63	**0.712**	0.668	0.678	0.671	**0.744**	0.731	0.487	0.472	**0.546**	0.494	0.511	0.503	**0.576**	**0.568**
SERA-NP-10	0.797	0.807	0.801	0.774	0.79	0.807	**0.818**	**0.843**	0.718	0.657	0.696	0.647	0.68	**0.731**	0.727	0.741	0.544	0.502	0.517	0.482	0.518	**0.552**	0.546	0.562
SERA-KW-5	0.783	0.778	0.804	0.808	0.786	0.775	0.806	0.796	0.594	0.622	0.652	0.684	0.697	0.65	0.696	0.648	0.424	0.468	0.502	0.507	0.525	0.478	0.523	0.482
SERA-KW-10	0.803	0.818	0.805	0.785	0.779	0.802	0.807	0.816	0.722	0.658	0.679	0.691	0.692	0.715	0.677	0.703	0.546	0.502	0.517	0.514	0.514	0.536	0.515	0.538
SERA-DIS-5	0.749	0.757	0.808	**0.822**	0.814	0.798	0.814	0.8	0.576	0.623	0.657	**0.697**	**0.73**	0.647	0.708	0.677	0.422	0.469	0.499	**0.522**	**0.562**	0.48	0.535	0.507
SERA-DIS-10	0.807	0.783	0.807	0.806	**0.815**	**0.813**	0.805	0.812	0.684	0.652	0.678	0.686	0.714	0.705	0.694	0.722	0.519	0.5	0.521	0.511	0.542	0.526	0.521	0.538
SERA-DIS-NP-5	0.781	0.736	0.815	0.803	0.805	0.799	0.805	0.806	0.629	0.644	0.7	0.65	0.697	0.661	0.703	0.727	0.479	0.475	0.542	0.482	0.523	0.499	0.525	0.548
SERA-DIS-NP-10	0.807	0.795	0.805	0.794	0.798	0.81	0.811	0.818	**0.732**	0.645	0.695	0.647	0.678	0.691	0.729	**0.743**	**0.569**	0.487	0.526	0.479	0.511	0.517	0.545	0.558
SERA-DIS-KW-5	0.748	0.751	0.806	0.813	0.804	0.788	0.807	0.789	0.575	0.61	0.664	0.686	0.707	0.624	0.704	0.656	0.42	0.456	0.509	0.506	0.537	0.46	0.529	0.487
SERA-DIS-KW-10	0.801	0.783	0.802	0.799	0.805	0.807	0.801	0.799	0.682	0.649	0.67	0.693	0.709	0.7	0.677	0.71	0.517	0.491	0.518	0.511	0.536	0.521	0.505	0.536
wikiSERA-5	0.783	0.779	**0.834**	0.816	0.789	0.778	0.809	0.81	0.57	0.615	**0.714**	0.687	0.669	0.645	0.688	0.681	0.423	0.456	**0.553**	0.515	0.514	0.481	0.528	0.513
wikiSERA-10	**0.797**	**0.816**	0.812	0.79	0.774	0.802	0.803	**0.832**	**0.713**	**0.648**	0.692	**0.702**	0.662	**0.732**	0.669	0.73	**0.545**	**0.494**	0.52	**0.523**	0.495	**0.549**	0.505	0.561
wikiSERA-DIS-5	0.76	0.758	0.83	**0.827**	**0.816**	0.807	**0.819**	0.815	0.581	0.623	0.689	0.693	**0.7**	0.667	**0.709**	0.686	0.422	0.464	0.525	0.521	**0.548**	0.494	**0.534**	0.518
wikiSERA-DIS-10	0.796	0.78	0.817	0.814	0.808	**0.818**	0.809	0.82	0.702	0.634	0.669	0.695	0.678	0.713	0.683	**0.743**	0.525	0.483	0.503	0.521	0.523	0.536	0.514	**0.563**

Table A.31: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.819	0.797	0.822	0.82	0.819	0.789	0.818	0.814	0.671	0.663	0.663	0.695	0.756	0.658	0.714	0.682	0.488	0.495	0.499	0.527	0.573	0.487	0.545	0.51
SERA-10	0.819	**0.833**	0.822	0.794	0.802	0.811	0.813	0.824	0.761	0.693	0.687	0.693	0.712	0.727	0.711	0.706	0.581	0.532	0.522	0.522	0.536	0.54	0.538	0.541
SERA-NP-5	0.818	0.781	**0.825**	0.798	0.794	0.789	**0.822**	0.84	0.7	0.647	0.7	0.67	0.696	0.702	**0.769**	0.74	0.525	0.48	0.538	0.504	0.526	0.534	**0.591**	0.569
SERA-NP-10	0.809	0.831	0.814	0.785	0.796	0.812	0.82	**0.841**	**0.773**	0.668	**0.718**	0.684	0.694	**0.739**	0.739	**0.765**	**0.584**	0.51	**0.542**	0.513	0.53	**0.566**	0.556	**0.588**
SERA-KW-5	0.815	0.791	0.814	0.813	0.812	0.781	0.813	0.806	0.687	0.656	0.665	**0.697**	0.749	0.656	0.727	0.692	0.509	0.484	0.503	**0.529**	0.568	0.48	0.554	0.505
SERA-KW-10	0.813	0.832	0.814	0.792	0.794	0.805	0.811	0.815	0.753	**0.698**	0.678	0.693	0.706	0.73	0.714	0.698	0.576	**0.538**	0.517	0.519	0.537	0.538	0.542	0.531
SERA-DIS-5	0.788	0.769	0.819	**0.826**	**0.832**	0.804	0.808	0.804	0.654	0.653	0.671	0.691	**0.762**	0.635	0.673	0.709	0.496	0.495	0.509	0.525	**0.596**	0.475	0.51	0.529
SERA-DIS-10	**0.821**	0.788	0.812	0.805	0.826	0.812	0.795	0.799	0.733	0.673	0.668	0.683	0.737	0.713	0.691	0.732	0.557	0.513	0.514	0.51	0.569	0.534	0.51	0.549
SERA-DIS-NP-5	0.783	0.743	0.815	0.806	0.817	0.816	0.811	0.819	0.66	0.646	0.697	0.683	0.719	0.718	0.725	0.729	0.501	0.486	0.534	0.517	0.554	0.54	0.554	0.553
SERA-DIS-NP-10	0.803	0.81	0.809	0.8	0.806	**0.818**	0.807	0.815	0.735	0.671	0.698	0.683	0.708	0.722	0.741	0.762	0.565	0.509	0.529	0.517	0.542	0.541	0.557	0.577
SERA-DIS-KW-5	0.788	0.763	0.815	0.816	0.823	0.792	0.802	0.791	0.646	0.646	0.677	0.687	0.755	0.623	0.687	0.686	0.488	0.484	0.514	0.514	0.587	0.461	0.514	0.506
SERA-DIS-KW-10	0.813	0.786	0.807	0.797	0.816	0.805	0.789	0.785	0.727	0.671	0.661	0.676	0.73	0.7	0.683	0.723	0.553	0.509	0.505	0.502	0.561	0.523	0.503	0.54
wikiSERA-5	0.799	0.795	**0.841**	0.824	0.813	0.786	0.813	0.826	0.627	0.656	0.696	**0.702**	0.732	0.66	0.698	0.727	0.449	0.489	0.529	**0.532**	0.554	0.482	0.53	0.553
wikiSERA-10	**0.815**	**0.827**	0.824	0.797	0.796	0.807	0.814	**0.833**	**0.739**	0.67	0.7	0.698	0.683	**0.737**	**0.708**	0.717	**0.566**	0.515	0.532	0.525	0.524	0.543	**0.533**	0.55
wikiSERA-DIS-5	0.778	0.77	0.836	**0.831**	**0.83**	0.808	**0.818**	0.825	0.624	0.669	**0.702**	0.701	**0.737**	0.666	0.695	0.741	0.447	0.506	**0.538**	0.529	**0.568**	0.496	0.528	0.569
wikiSERA-DIS-10	0.809	0.783	0.825	0.815	0.822	**0.817**	0.807	0.808	0.733	**0.68**	0.695	**0.702**	0.728	0.727	0.704	**0.759**	0.555	**0.527**	0.53	0.53	0.565	**0.544**	0.532	**0.579**

Table A.32: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.791	0.764	**0.805**	0.806	0.8	0.771	0.803	0.799	0.589	0.571	0.621	0.682	**0.723**	0.606	0.669	0.659	0.433	0.434	0.464	**0.502**	0.546	0.438	0.499	0.481
SERA-10	**0.819**	**0.814**	**0.805**	0.775	0.794	**0.803**	**0.812**	0.821	0.741	**0.649**	0.657	0.655	0.715	0.714	0.684	0.713	**0.572**	**0.507**	0.495	0.482	0.54	**0.542**	0.513	0.548
SERA-NP-5	0.81	0.743	**0.805**	0.787	0.794	0.783	**0.812**	0.825	0.666	0.592	0.671	0.646	0.691	0.676	**0.738**	**0.73**	0.496	0.43	0.497	0.479	0.524	0.515	**0.563**	**0.558**
SERA-NP-10	0.805	0.803	0.802	0.775	0.778	0.799	**0.812**	**0.828**	**0.747**	0.631	**0.68**	0.636	0.627	**0.723**	0.722	0.716	0.563	0.492	**0.515**	0.464	0.468	0.541	0.544	0.54
SERA-KW-5	0.791	0.76	0.798	0.801	0.794	0.765	0.797	0.794	0.6	0.571	0.617	**0.684**	0.717	0.599	0.674	0.667	0.446	0.423	0.464	0.501	0.539	0.437	0.506	0.495
SERA-KW-10	0.812	0.813	0.8	0.774	0.788	0.798	**0.812**	0.814	0.719	0.64	0.657	0.66	0.703	0.712	0.688	0.712	0.55	0.496	0.494	0.486	0.525	0.538	0.519	0.549
SERA-DIS-5	0.755	0.738	0.801	**0.809**	**0.816**	0.781	0.795	0.786	0.525	0.558	0.63	0.667	0.715	0.596	0.654	0.669	0.389	0.424	0.471	0.494	**0.563**	0.441	0.483	0.499
SERA-DIS-10	0.805	0.767	0.797	0.788	0.813	0.796	0.785	0.786	0.673	0.61	0.646	0.656	0.708	0.679	0.668	0.723	0.513	0.463	0.494	0.491	0.541	0.504	0.495	0.544
SERA-DIS-NP-5	0.779	0.711	0.797	0.795	0.812	0.795	0.801	0.798	0.581	0.584	0.661	0.629	0.7	0.669	0.671	0.7	0.433	0.43	0.509	0.471	0.529	0.498	0.505	0.526
SERA-DIS-NP-10	0.796	0.787	0.801	0.79	0.791	0.795	0.792	0.79	0.686	0.611	0.664	0.657	0.666	0.696	0.684	0.698	0.523	0.473	0.501	0.49	0.508	0.505	0.502	0.523
SERA-DIS-KW-5	0.76	0.735	0.799	0.802	0.806	0.771	0.788	0.779	0.534	0.55	0.64	0.673	0.704	0.576	0.645	0.645	0.393	0.406	0.486	0.491	0.546	0.425	0.48	0.478
SERA-DIS-KW-10	0.799	0.766	0.795	0.784	0.802	0.788	0.782	0.774	0.662	0.609	0.642	0.661	0.699	0.656	0.667	0.729	0.492	0.463	0.487	0.494	0.53	0.482	0.503	0.548
wikiSERA-5	0.772	0.762	**0.824**	0.81	0.8	0.764	0.807	0.811	0.542	0.602	**0.671**	**0.679**	0.69	0.585	**0.684**	0.697	0.397	0.451	**0.513**	**0.509**	0.522	0.441	**0.522**	0.528
wikiSERA-10	**0.81**	**0.81**	0.806	0.781	0.785	**0.805**	0.808	**0.827**	**0.731**	**0.627**	0.662	0.669	0.669	**0.727**	0.682	0.715	**0.553**	**0.49**	0.495	0.496	0.501	**0.546**	0.508	**0.55**
wikiSERA-DIS-5	0.755	0.74	0.82	**0.812**	**0.817**	0.79	**0.81**	0.808	0.527	0.602	0.645	0.662	**0.71**	0.612	0.675	0.684	0.38	0.449	0.486	0.499	**0.545**	0.456	0.507	0.519
wikiSERA-DIS-10	0.792	0.763	0.807	0.795	0.809	**0.805**	0.794	0.794	0.659	0.608	0.653	0.656	0.69	0.683	0.68	**0.726**	0.482	0.465	0.492	0.487	0.533	0.506	0.514	0.546

Table A.33: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

	Pearson								Spearman								Kendall							
Method	TAC2009	AQUAINT-2							TAC2009	AQUAINT-2							TAC2009	AQUAINT-2						
		825,148	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**		**825,148**	**179,520**	**89,760**	**60,000**	**30,000**	**15,000**	**10,000**
SERA-5	0.799	0.782	0.814	0.812	0.803	0.782	0.812	0.807	0.607	0.628	0.664	**0.687**	0.712	0.641	0.698	0.665	0.439	0.468	0.502	**0.515**	0.538	0.472	0.533	0.494
SERA-10	0.812	**0.822**	0.813	0.784	0.793	0.808	0.813	0.824	**0.74**	0.662	0.675	0.683	0.703	0.722	0.69	0.705	**0.564**	0.509	0.513	0.514	0.528	0.54	0.518	0.541
SERA-NP-5	**0.814**	0.766	**0.816**	0.794	0.793	0.781	0.818	0.837	0.671	0.625	0.692	0.662	0.682	0.671	**0.753**	0.734	0.504	0.466	**0.53**	0.49	0.521	0.513	**0.578**	0.558
SERA-NP-10	0.802	0.816	0.806	0.778	0.787	0.807	**0.819**	**0.839**	**0.74**	0.626	**0.702**	0.649	0.651	**0.733**	0.735	**0.75**	0.552	0.484	0.529	0.482	0.498	**0.554**	0.557	**0.57**
SERA-KW-5	0.797	0.776	0.807	0.807	0.796	0.775	0.808	0.801	0.62	0.624	0.665	0.681	0.711	0.629	0.716	0.672	0.456	0.465	0.503	0.51	0.537	0.462	0.543	0.495
SERA-KW-10	0.807	0.821	0.807	0.783	0.787	0.802	0.812	0.817	0.732	**0.665**	0.67	**0.687**	0.7	0.725	0.687	0.707	0.56	**0.516**	0.511	0.513	0.529	0.544	0.519	0.542
SERA-DIS-5	0.764	0.755	0.811	**0.818**	**0.821**	0.795	0.809	0.801	0.583	0.615	0.66	0.677	**0.732**	0.629	0.699	0.689	0.428	0.459	0.501	0.506	**0.57**	0.457	0.519	0.514
SERA-DIS-10	0.81	0.779	0.806	0.798	0.818	0.807	0.798	0.802	0.693	0.636	0.673	0.666	0.712	0.699	0.692	0.725	0.523	0.484	0.521	0.492	0.545	0.525	0.517	0.544
SERA-DIS-NP-5	0.781	0.733	0.809	0.804	0.812	0.803	0.81	0.815	0.629	0.638	0.687	0.664	0.707	0.685	0.722	0.728	0.484	0.474	0.529	0.502	0.532	0.52	0.545	0.55
SERA-DIS-NP-10	0.801	0.782	0.806	0.796	0.799	**0.81**	0.805	0.811	0.719	0.639	0.687	0.67	0.679	0.711	0.732	0.736	0.556	0.487	0.518	0.505	0.514	0.53	0.546	0.549
SERA-DIS-KW-5	0.766	0.75	0.808	0.81	0.811	0.785	0.804	0.791	0.564	0.6	0.653	0.681	0.726	0.623	0.7	0.671	0.414	0.441	0.496	0.502	0.56	0.457	0.523	0.502
SERA-DIS-KW-10	0.804	0.777	0.802	0.792	0.808	0.8	0.795	0.79	0.694	0.632	0.669	0.68	0.705	0.685	0.682	0.726	0.523	0.482	0.515	0.507	0.537	0.509	0.514	0.545
wikiSERA-5	0.783	0.779	**0.834**	0.816	0.799	0.776	0.812	0.817	0.583	0.631	**0.697**	0.683	0.682	0.638	0.697	0.698	0.429	0.47	**0.53**	0.511	0.521	0.477	0.535	0.527
wikiSERA-10	**0.805**	**0.818**	0.814	0.789	0.785	0.805	0.81	**0.832**	**0.711**	**0.652**	0.686	0.686	0.672	**0.728**	0.692	0.724	**0.54**	**0.502**	0.517	0.51	0.511	**0.541**	0.519	0.557
wikiSERA-DIS-5	0.763	0.756	0.831	**0.825**	**0.82**	0.802	**0.82**	0.819	0.578	0.628	0.684	**0.688**	**0.714**	0.655	**0.707**	0.698	0.41	0.469	0.525	**0.518**	**0.55**	0.483	**0.538**	0.522
wikiSERA-DIS-10	0.801	0.775	0.818	0.81	0.814	**0.815**	0.807	0.81	0.705	0.635	0.673	0.686	0.709	0.719	0.706	**0.745**	0.527	0.485	0.514	0.511	**0.55**	0.54	0.534	**0.565**

Table A.34: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/AQUAINT-2 dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.5 Correlation of SERA and wikiSERA with Pyramid on TAC2008/Wikipedia

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.807	0.802	0.854	0.807	0.773	0.822	0.874	0.845	0.58	0.639	0.697	0.658
	SERA-10	0.821	0.877	0.864	0.848	0.804	0.9	0.887	0.873	0.612	0.728	0.717	0.692
	SERA-NP-5	**0.826**	0.88	0.858	0.807	**0.806**	0.874	0.874	0.824	**0.615**	0.696	0.687	0.63
	SERA-NP-10	0.821	0.853	0.862	0.822	0.771	0.878	0.885	0.852	0.59	0.703	0.721	0.671
	SERA-KW-5	0.769	0.813	0.85	0.802	0.732	0.83	0.863	0.843	0.542	0.648	0.684	0.652
	SERA-KW-10	0.79	0.874	0.872	0.848	0.749	0.895	**0.891**	0.877	0.555	0.723	0.719	0.69
	SERA-DIS-5	0.794	0.807	0.851	0.818	0.775	0.828	0.881	0.876	0.584	0.648	0.702	0.691
	SERA-DIS-10	0.786	0.876	0.854	**0.851**	0.77	**0.908**	0.87	**0.888**	0.577	0.733	0.684	**0.711**
	SERA-DIS-NP-5	0.789	**0.885**	0.86	0.829	0.759	0.856	0.875	0.827	0.562	0.682	0.69	0.637
	SERA-DIS-NP-10	0.799	0.881	**0.874**	0.85	0.75	0.904	**0.891**	0.869	0.567	0.73	**0.725**	0.684
	SERA-DIS-KW-5	0.744	0.821	0.852	0.805	0.726	0.834	0.878	0.855	0.532	0.65	0.699	0.667
	SERA-DIS-KW-10	0.762	0.883	0.865	0.846	0.746	**0.908**	0.874	0.882	0.55	**0.736**	0.682	0.701
	wikiSERA-5	0.79	0.849	0.851	0.819	0.767	0.866	0.853	0.833	0.565	0.691	0.679	0.651
	wikiSERA-10	**0.828**	**0.874**	**0.869**	0.826	0.781	**0.894**	**0.876**	0.845	0.592	**0.717**	**0.706**	0.656
	wikiSERA-DIS-5	0.783	0.833	0.839	0.826	0.784	0.847	0.836	0.849	0.584	0.67	0.662	0.668
	wikiSERA-DIS-10	0.818	0.867	0.864	**0.837**	**0.797**	0.89	0.861	**0.865**	**0.607**	0.713	0.693	**0.682**
		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.807	0.802	0.854	0.807	0.771	0.82	0.873	0.843	0.58	0.637	0.694	0.656
	SERA-10	0.82	0.876	0.864	0.848	0.801	0.9	0.885	0.87	0.61	0.725	0.714	0.688
	SERA-NP-5	**0.826**	0.88	0.857	0.807	**0.806**	0.874	0.874	0.823	**0.615**	0.696	0.687	0.628
	SERA-NP-10	0.821	0.853	0.861	0.821	0.769	0.877	0.882	0.848	0.588	0.701	0.718	0.668
	SERA-KW-5	0.768	0.814	0.851	0.802	0.729	0.829	0.862	0.839	0.54	0.645	0.682	0.649
	SERA-KW-10	0.786	0.874	0.871	0.848	0.749	0.895	**0.889**	0.874	0.553	0.721	0.717	0.688
	SERA-DIS-5	0.793	0.807	0.85	0.818	0.774	0.825	0.879	0.873	0.581	0.645	0.7	0.689
	SERA-DIS-10	0.785	0.876	0.854	**0.851**	0.767	**0.907**	0.869	**0.884**	0.575	0.73	0.681	**0.708**
	SERA-DIS-NP-5	0.789	**0.885**	0.86	0.829	0.758	0.855	0.874	0.826	0.56	0.679	0.688	0.635
	SERA-DIS-NP-10	0.798	0.881	**0.873**	0.85	0.748	0.903	**0.889**	0.866	0.564	0.728	**0.723**	0.682
	SERA-DIS-KW-5	0.743	0.821	0.852	0.805	0.722	0.832	0.877	0.852	0.527	0.648	0.696	0.665
	SERA-DIS-KW-10	0.759	0.883	0.864	0.846	0.74	**0.907**	0.872	0.878	0.544	**0.734**	0.679	0.699
	wikiSERA-5	0.789	0.85	0.85	0.819	0.764	0.865	0.852	0.831	0.56	0.688	0.677	0.649
	wikiSERA-10	**0.827**	**0.874**	**0.869**	0.826	0.779	**0.894**	**0.875**	0.843	0.59	**0.717**	**0.704**	0.656
	wikiSERA-DIS-5	0.781	0.833	0.839	0.826	0.78	0.844	0.835	0.847	0.579	0.667	0.66	0.666
	wikiSERA-DIS-10	0.817	0.867	0.864	**0.836**	**0.795**	0.889	0.86	**0.863**	**0.604**	0.711	0.69	**0.682**

Table A.35: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.808	0.831	0.859	0.817	0.772	0.811	0.834	0.789	0.571	0.633	0.648	0.599
	SERA-10	0.833	0.887	0.876	0.842	0.777	0.877	0.889	0.843	0.594	0.708	0.706	0.654
	SERA-NP-5	0.818	**0.91**	0.854	0.798	0.787	0.896	0.854	0.776	0.59	0.721	0.673	0.579
	SERA-NP-10	**0.839**	0.882	0.861	**0.855**	**0.844**	0.885	0.839	**0.867**	**0.648**	0.698	0.649	**0.672**
	SERA-KW-5	0.805	0.846	0.857	0.815	0.758	0.811	0.828	0.794	0.564	0.639	0.641	0.606
	SERA-KW-10	0.825	0.885	**0.885**	0.844	0.773	0.87	**0.891**	0.852	0.582	0.693	**0.707**	0.66
	SERA-DIS-5	0.808	0.825	0.848	0.825	0.772	0.798	0.813	0.781	0.574	0.609	0.621	0.59
	SERA-DIS-10	0.805	0.871	0.871	0.84	0.749	0.853	0.856	0.826	0.554	0.668	0.661	0.637
	SERA-DIS-NP-5	0.777	0.892	0.868	0.814	0.76	0.865	0.852	0.771	0.555	0.665	0.653	0.573
	SERA-DIS-NP-10	0.821	0.901	0.872	0.846	0.819	**0.9**	0.857	0.818	0.625	**0.724**	0.671	0.624
	SERA-DIS-KW-5	0.759	0.842	0.851	0.826	0.749	0.815	0.808	0.792	0.549	0.632	0.621	0.606
	SERA-DIS-KW-10	0.775	0.872	0.876	0.843	0.762	0.853	0.855	0.821	0.565	0.671	0.666	0.633
	wikiSERA-5	0.813	0.89	**0.867**	0.83	0.805	0.871	0.857	0.841	0.624	0.716	0.671	0.656
	wikiSERA-10	**0.863**	0.891	0.866	0.858	**0.834**	0.891	**0.867**	**0.859**	**0.652**	0.718	**0.683**	**0.673**
	wikiSERA-DIS-5	0.774	0.877	0.851	0.826	0.752	0.861	0.821	0.807	0.558	0.685	0.631	0.624
	wikiSERA-DIS-10	0.833	**0.892**	0.865	**0.862**	0.805	**0.904**	0.861	0.846	0.615	**0.739**	0.669	0.653

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.807	0.83	0.858	0.816	0.769	0.808	0.832	0.785	0.568	0.631	0.646	0.596
	SERA-10	0.832	0.886	0.875	0.841	0.775	0.875	0.887	0.841	0.594	0.706	0.704	0.652
	SERA-NP-5	0.818	**0.909**	0.853	0.797	0.786	0.896	0.853	0.773	0.59	0.721	0.671	0.577
	SERA-NP-10	**0.839**	0.882	0.86	**0.854**	**0.842**	0.884	0.837	**0.864**	**0.646**	0.698	0.647	**0.669**
	SERA-KW-5	0.804	0.845	0.856	0.814	0.757	0.808	0.826	0.79	0.564	0.636	0.639	0.603
	SERA-KW-10	0.821	0.884	**0.884**	0.843	0.777	0.868	**0.889**	0.85	0.582	0.691	**0.705**	0.658
	SERA-DIS-5	0.808	0.824	0.847	0.824	0.77	0.794	0.81	0.777	0.572	0.607	0.619	0.587
	SERA-DIS-10	0.804	0.87	0.871	0.839	0.747	0.85	0.853	0.823	0.551	0.666	0.659	0.635
	SERA-DIS-NP-5	0.776	0.892	0.868	0.814	0.758	0.865	0.852	0.768	0.555	0.662	0.65	0.57
	SERA-DIS-NP-10	0.82	0.9	0.872	0.845	0.816	**0.899**	0.855	0.815	0.623	**0.724**	0.668	0.621
	SERA-DIS-KW-5	0.757	0.842	0.85	0.825	0.748	0.811	0.806	0.788	0.546	0.63	0.619	0.603
	SERA-DIS-KW-10	0.772	0.871	0.876	0.842	0.763	0.849	0.852	0.819	0.558	0.668	0.664	0.631
	wikiSERA-5	0.812	0.889	**0.866**	0.828	0.802	0.868	0.855	0.837	0.622	0.713	0.669	0.651
	wikiSERA-10	**0.862**	0.891	**0.866**	0.858	**0.832**	0.89	**0.865**	**0.858**	**0.649**	0.715	**0.68**	**0.671**
	wikiSERA-DIS-5	0.773	0.877	0.85	0.824	0.75	0.859	0.818	0.802	0.556	0.683	0.629	0.621
	wikiSERA-DIS-10	0.832	**0.892**	0.865	**0.862**	0.803	**0.902**	0.858	0.843	0.613	**0.736**	0.667	0.65

Table A.36: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_2$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.804	0.747	0.825	0.813	0.755	0.764	0.837	0.829	0.578	0.577	0.643	0.629
	SERA-10	0.829	0.835	**0.848**	0.851	0.784	0.852	**0.891**	0.884	0.601	0.673	**0.695**	**0.7**
	SERA-NP-5	0.819	0.811	0.828	0.742	0.757	0.83	0.837	0.742	0.587	0.65	0.639	0.554
	SERA-NP-10	**0.831**	0.862	0.84	0.836	**0.804**	**0.896**	0.863	**0.888**	**0.624**	**0.709**	0.668	0.699
	SERA-KW-5	0.795	0.75	0.823	0.812	0.759	0.765	0.83	0.825	0.581	0.577	0.63	0.625
	SERA-KW-10	0.812	0.839	**0.848**	**0.852**	0.775	0.862	0.885	0.88	0.599	0.682	0.693	0.699
	SERA-DIS-5	0.806	0.751	0.794	0.793	0.75	0.751	0.806	0.822	0.569	0.563	0.602	0.621
	SERA-DIS-10	0.82	0.82	0.833	0.836	0.785	0.828	0.856	0.867	0.598	0.633	0.659	0.677
	SERA-DIS-NP-5	0.788	0.818	0.799	0.705	0.753	0.817	0.83	0.707	0.58	0.632	0.638	0.529
	SERA-DIS-NP-10	0.804	**0.865**	0.823	0.793	0.787	0.886	0.844	0.838	0.606	0.696	0.646	0.643
	SERA-DIS-KW-5	0.793	0.742	0.791	0.783	0.761	0.728	0.788	0.8	0.568	0.537	0.59	0.602
	SERA-DIS-KW-10	0.812	0.819	0.832	0.825	0.774	0.834	0.849	0.859	0.583	0.638	0.652	0.67
	wikiSERA-5	0.815	0.803	0.835	0.803	**0.812**	0.815	0.849	0.851	**0.626**	0.635	0.678	0.665
	wikiSERA-10	**0.83**	**0.838**	**0.859**	0.838	0.77	**0.856**	**0.875**	**0.859**	0.588	**0.672**	**0.694**	**0.677**
	wikiSERA-DIS-5	0.809	0.8	0.793	0.808	0.809	0.779	0.802	0.852	0.619	0.597	0.621	0.667
	wikiSERA-DIS-10	0.824	0.827	0.835	**0.839**	0.787	0.836	0.837	0.854	0.606	0.655	0.66	0.675

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.804	0.748	0.825	0.813	0.755	0.763	0.835	0.826	0.578	0.575	0.641	0.627
	SERA-10	0.828	0.835	**0.848**	0.851	0.782	0.851	**0.889**	0.882	0.601	0.67	**0.692**	**0.698**
	SERA-NP-5	0.818	0.812	0.828	0.743	0.756	0.829	0.838	0.74	0.584	0.648	0.642	0.551
	SERA-NP-10	**0.831**	0.862	0.839	0.836	**0.803**	**0.895**	0.861	**0.886**	**0.622**	**0.707**	0.666	0.696
	SERA-KW-5	0.794	0.751	0.822	0.811	0.758	0.765	0.829	0.823	0.581	0.575	0.628	0.622
	SERA-KW-10	0.811	0.839	**0.848**	**0.852**	0.771	0.861	0.883	0.877	0.596	0.679	0.69	0.697
	SERA-DIS-5	0.805	0.751	0.794	0.792	0.749	0.751	0.805	0.819	0.567	0.561	0.599	0.619
	SERA-DIS-10	0.82	0.819	0.833	0.835	0.782	0.827	0.855	0.864	0.598	0.631	0.656	0.675
	SERA-DIS-NP-5	0.787	0.819	0.799	0.705	0.752	0.816	0.83	0.704	0.578	0.63	0.636	0.527
	SERA-DIS-NP-10	0.803	**0.866**	0.823	0.793	0.785	0.885	0.842	0.836	0.603	0.694	0.644	0.641
	SERA-DIS-KW-5	0.792	0.742	0.791	0.782	0.758	0.727	0.787	0.797	0.568	0.534	0.587	0.6
	SERA-DIS-KW-10	0.81	0.819	0.832	0.825	0.775	0.833	0.847	0.856	0.579	0.636	0.649	0.667
	wikiSERA-5	0.814	0.803	0.835	0.802	**0.808**	0.814	0.849	0.849	**0.623**	0.633	0.678	0.663
	wikiSERA-10	**0.829**	**0.838**	**0.859**	0.838	0.767	**0.856**	**0.874**	**0.857**	0.585	**0.67**	**0.691**	**0.674**
	wikiSERA-DIS-5	0.808	0.8	0.793	0.807	0.806	0.778	0.801	0.849	0.616	0.597	0.621	0.665
	wikiSERA-DIS-10	0.823	0.827	0.835	**0.839**	0.783	0.836	0.835	0.852	0.603	0.653	0.657	0.672

Table A.37: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_3$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.806	0.817	0.869	0.799	0.76	0.81	0.863	0.824	0.589	0.632	0.689	0.634
	SERA-10	0.834	0.861	0.887	0.858	0.774	0.867	**0.9**	0.864	0.59	0.69	**0.726**	0.679
	SERA-NP-5	0.857	0.866	0.854	0.804	0.81	0.859	0.883	0.815	0.608	0.692	0.696	0.627
	SERA-NP-10	0.811	0.87	0.856	0.855	0.772	**0.882**	0.876	0.854	0.588	**0.704**	0.7	0.663
	SERA-KW-5	0.831	0.809	0.869	0.792	0.802	0.812	0.871	0.813	0.617	0.634	0.7	0.613
	SERA-KW-10	0.85	0.869	0.884	0.856	0.82	0.879	0.888	**0.873**	0.63	0.703	0.714	**0.687**
	SERA-DIS-5	0.828	0.829	0.858	0.796	0.764	0.803	0.858	0.813	0.581	0.619	0.682	0.623
	SERA-DIS-10	0.825	0.864	**0.888**	0.856	0.777	0.852	0.896	0.863	0.593	0.676	0.714	0.681
	SERA-DIS-NP-5	0.847	0.843	0.84	0.816	0.814	0.803	0.871	0.813	0.624	0.627	0.678	0.614
	SERA-DIS-NP-10	0.831	**0.873**	0.846	**0.86**	0.791	0.88	0.875	0.846	0.591	0.7	0.69	0.66
	SERA-DIS-KW-5	0.848	0.821	0.864	0.788	0.821	0.801	0.859	0.812	0.638	0.619	0.681	0.607
	SERA-DIS-KW-10	**0.865**	0.866	0.882	0.854	**0.839**	0.864	0.885	0.854	**0.656**	0.684	0.706	0.659
	wikiSERA-5	0.849	0.846	0.865	0.822	0.831	0.836	**0.885**	0.845	0.645	0.648	**0.701**	0.664
	wikiSERA-10	0.842	0.858	**0.871**	0.855	0.792	**0.876**	0.852	**0.851**	0.609	0.691	0.67	**0.671**
	wikiSERA-DIS-5	**0.853**	0.85	0.839	0.816	**0.839**	0.838	0.857	0.823	**0.656**	0.658	0.668	0.638
	wikiSERA-DIS-10	0.849	**0.86**	0.86	**0.856**	0.781	**0.876**	0.848	0.85	0.598	**0.693**	0.66	**0.671**
		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.806	0.817	0.869	0.799	0.758	0.808	0.862	0.82	0.587	0.632	0.689	0.632
	SERA-10	0.834	0.861	0.887	0.858	0.772	0.866	**0.897**	0.861	0.59	0.69	**0.723**	0.677
	SERA-NP-5	0.856	0.866	0.854	0.804	0.807	0.857	0.881	0.814	0.605	0.69	0.693	0.624
	SERA-NP-10	0.811	0.87	0.855	0.855	0.77	**0.88**	0.874	0.851	0.586	**0.702**	0.697	0.66
	SERA-KW-5	0.83	0.81	0.87	0.792	0.799	0.809	0.869	0.81	0.617	0.634	0.698	0.611
	SERA-KW-10	0.848	0.869	0.884	0.856	0.816	0.877	0.886	**0.871**	0.63	0.701	0.712	**0.684**
	SERA-DIS-5	0.828	0.829	0.858	0.796	0.76	0.802	0.858	0.811	0.579	0.619	0.682	0.62
	SERA-DIS-10	0.824	0.864	**0.888**	0.856	0.775	0.851	0.895	0.86	0.593	0.676	0.714	0.678
	SERA-DIS-NP-5	0.846	0.843	0.84	0.816	0.81	0.802	0.87	0.813	0.621	0.625	0.676	0.614
	SERA-DIS-NP-10	0.83	**0.873**	0.846	**0.859**	0.788	0.878	0.873	0.844	0.589	0.698	0.688	0.658
	SERA-DIS-KW-5	0.847	0.821	0.864	0.788	0.817	0.799	0.859	0.81	0.638	0.619	0.681	0.604
	SERA-DIS-KW-10	**0.867**	0.866	0.882	0.855	**0.835**	0.863	0.884	0.851	**0.656**	0.684	0.706	0.656
	wikiSERA-5	0.848	0.847	0.865	0.821	0.828	0.832	**0.884**	0.843	0.643	0.646	**0.699**	0.662
	wikiSERA-10	0.841	0.859	**0.871**	0.855	0.789	**0.875**	0.85	**0.849**	0.609	0.688	0.67	**0.668**
	wikiSERA-DIS-5	**0.852**	0.85	0.839	0.816	**0.835**	0.835	0.857	0.821	**0.654**	0.655	0.668	0.636
	wikiSERA-DIS-10	0.849	**0.86**	0.86	**0.856**	0.778	**0.875**	0.847	**0.849**	0.595	**0.69**	0.66	**0.668**

Table A.38: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_4$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.824	0.822	0.871	0.843	0.79	0.832	0.881	0.843	0.603	0.65	0.7	0.657
	SERA-10	0.844	0.884	0.878	0.866	0.798	0.906	**0.906**	**0.891**	0.614	0.737	0.731	**0.713**
	SERA-NP-5	0.841	**0.902**	0.88	0.819	0.805	0.898	0.887	0.818	0.607	0.726	0.7	0.623
	SERA-NP-10	**0.85**	0.885	0.873	0.861	**0.833**	0.904	0.888	0.884	**0.648**	0.72	0.711	0.705
	SERA-KW-5	0.81	0.833	0.868	0.842	0.777	0.844	0.875	0.853	0.587	0.669	0.689	0.667
	SERA-KW-10	0.829	0.884	**0.884**	0.866	0.794	0.91	**0.906**	0.888	0.606	0.737	**0.734**	0.706
	SERA-DIS-5	0.828	0.823	0.859	0.847	0.793	0.819	0.864	0.863	0.606	0.643	0.678	0.676
	SERA-DIS-10	0.82	0.874	0.872	**0.869**	0.789	0.898	0.872	0.89	0.6	0.724	0.682	0.704
	SERA-DIS-NP-5	0.806	0.896	0.875	0.819	0.781	0.891	0.873	0.82	0.589	0.716	0.688	0.641
	SERA-DIS-NP-10	0.83	0.901	0.879	0.851	0.814	**0.915**	0.884	0.873	0.624	**0.748**	0.699	0.683
	SERA-DIS-KW-5	0.789	0.832	0.86	0.842	0.776	0.829	0.864	0.856	0.585	0.652	0.681	0.668
	SERA-DIS-KW-10	0.807	0.879	0.879	0.864	0.792	0.898	0.879	0.878	0.597	0.724	0.696	0.695
	wikiSERA-5	0.828	0.872	0.881	0.854	**0.814**	0.868	0.887	0.868	0.62	0.693	0.713	0.691
	wikiSERA-10	**0.855**	**0.884**	**0.882**	0.861	**0.814**	**0.909**	**0.891**	0.875	**0.625**	**0.737**	**0.719**	**0.699**
	wikiSERA-DIS-5	0.816	0.865	0.863	0.854	0.811	0.861	0.857	0.87	0.616	0.69	0.671	0.69
	wikiSERA-DIS-10	0.842	0.88	0.876	**0.867**	0.809	0.895	0.87	**0.881**	0.624	0.719	0.689	0.696

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.823	0.822	0.871	0.843	0.79	0.83	0.879	0.84	0.603	0.648	0.698	0.655
	SERA-10	0.843	0.884	0.878	0.865	0.796	0.906	**0.904**	**0.888**	0.614	0.735	0.729	**0.71**
	SERA-NP-5	0.84	**0.902**	0.88	0.819	0.804	0.898	0.887	0.816	0.604	0.724	0.7	0.62
	SERA-NP-10	**0.85**	0.885	0.873	0.861	**0.831**	0.904	0.885	0.881	**0.645**	0.718	0.709	0.702
	SERA-KW-5	0.809	0.833	0.867	0.841	0.775	0.842	0.873	0.849	0.587	0.666	0.687	0.665
	SERA-KW-10	0.827	0.884	**0.884**	0.865	0.792	0.909	**0.904**	0.885	0.601	0.735	**0.731**	0.704
	SERA-DIS-5	0.827	0.823	0.859	0.847	0.791	0.817	0.862	0.859	0.603	0.641	0.676	0.673
	SERA-DIS-10	0.819	0.874	0.872	**0.868**	0.787	0.896	0.87	0.887	0.597	0.722	0.679	0.701
	SERA-DIS-NP-5	0.805	0.896	0.875	0.819	0.78	0.89	0.872	0.818	0.587	0.713	0.685	0.638
	SERA-DIS-NP-10	0.829	0.901	0.879	0.851	0.811	**0.914**	0.881	0.87	0.621	**0.746**	0.696	0.681
	SERA-DIS-KW-5	0.788	0.832	0.86	0.842	0.774	0.827	0.862	0.853	0.582	0.649	0.678	0.666
	SERA-DIS-KW-10	0.806	0.878	0.879	0.863	0.79	0.897	0.877	0.875	0.599	0.722	0.694	0.693
	wikiSERA-5	0.827	0.872	0.88	0.853	0.811	0.865	0.887	0.866	0.617	0.691	0.71	0.689
	wikiSERA-10	**0.854**	**0.884**	**0.882**	0.86	**0.812**	**0.909**	**0.889**	0.873	**0.623**	**0.735**	**0.716**	**0.696**
	wikiSERA-DIS-5	0.815	0.865	0.863	0.854	0.807	0.858	0.855	0.866	0.614	0.688	0.668	0.688
	wikiSERA-DIS-10	0.841	0.88	0.875	**0.867**	0.807	0.894	0.868	**0.878**	0.621	0.716	0.687	0.694

Table A.39: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.827	0.844	0.886	0.841	0.78	0.839	0.89	0.84	0.591	0.666	0.716	0.652
	SERA-10	0.845	0.894	0.89	0.87	0.797	0.903	0.909	0.888	0.61	0.725	0.734	0.712
	SERA-NP-5	**0.857**	**0.909**	0.883	0.838	0.828	0.897	0.901	0.84	0.631	0.727	0.725	0.654
	SERA-NP-10	0.842	0.891	0.879	0.867	**0.831**	0.908	0.89	0.877	**0.647**	0.735	0.719	0.696
	SERA-KW-5	0.823	0.852	0.884	0.836	0.797	0.852	0.893	0.847	0.61	0.682	0.717	0.66
	SERA-KW-10	0.842	0.894	**0.895**	0.87	0.814	0.908	**0.91**	0.887	0.628	0.74	**0.737**	0.707
	SERA-DIS-5	0.834	0.845	0.88	0.854	0.786	0.836	0.874	0.86	0.602	0.659	0.702	0.678
	SERA-DIS-10	0.823	0.888	0.889	0.875	0.779	0.901	0.885	**0.894**	0.591	0.724	0.704	**0.718**
	SERA-DIS-NP-5	0.83	0.9	0.886	0.858	0.807	0.879	0.897	0.843	0.606	0.707	0.721	0.66
	SERA-DIS-NP-10	0.838	0.905	0.888	0.874	0.815	**0.912**	0.9	0.881	0.627	**0.747**	0.723	0.694
	SERA-DIS-KW-5	0.806	0.855	0.884	0.849	0.785	0.839	0.882	0.849	0.595	0.667	0.707	0.665
	SERA-DIS-KW-10	0.825	0.892	0.894	**0.876**	0.805	0.906	0.893	0.891	0.606	0.734	0.714	0.706
	wikiSERA-5	0.839	0.884	**0.89**	0.856	**0.83**	0.881	**0.902**	0.864	0.633	0.71	**0.731**	0.677
	wikiSERA-10	**0.862**	0.891	0.885	0.865	0.824	**0.909**	0.891	0.87	**0.642**	0.734	0.714	0.688
	wikiSERA-DIS-5	0.826	0.879	0.878	0.857	0.821	0.872	0.875	0.863	0.629	0.704	0.694	0.681
	wikiSERA-DIS-10	0.849	**0.892**	0.885	**0.875**	0.817	**0.909**	0.876	**0.873**	0.629	**0.742**	0.695	**0.694**

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.826	0.844	0.886	0.841	0.778	0.837	0.888	0.836	0.589	0.664	0.713	0.649
	SERA-10	0.844	0.894	0.889	0.87	0.795	0.902	0.906	0.885	0.61	0.723	0.731	0.71
	SERA-NP-5	**0.856**	**0.909**	0.883	0.837	0.827	0.896	0.899	0.838	0.628	0.724	0.723	0.652
	SERA-NP-10	0.842	0.89	0.878	0.867	**0.828**	0.907	0.887	0.873	**0.644**	0.733	0.717	0.694
	SERA-KW-5	0.822	0.852	0.884	0.836	0.795	0.849	0.891	0.844	0.608	0.679	0.715	0.658
	SERA-KW-10	0.839	0.894	**0.895**	0.87	0.807	0.907	**0.907**	0.884	0.625	0.737	**0.735**	0.705
	SERA-DIS-5	0.833	0.845	0.88	0.854	0.783	0.834	0.872	0.857	0.6	0.656	0.7	0.676
	SERA-DIS-10	0.822	0.888	0.889	0.875	0.776	0.899	0.883	**0.892**	0.589	0.722	0.701	**0.716**
	SERA-DIS-NP-5	0.829	0.9	0.886	0.858	0.804	0.878	0.897	0.842	0.604	0.705	0.718	0.657
	SERA-DIS-NP-10	0.837	0.905	0.888	0.873	0.812	**0.911**	0.898	0.878	0.625	**0.745**	0.721	0.691
	SERA-DIS-KW-5	0.805	0.855	0.884	0.849	0.781	0.836	0.88	0.846	0.59	0.665	0.705	0.662
	SERA-DIS-KW-10	0.825	0.892	0.894	**0.876**	0.8	0.904	0.89	0.888	0.606	0.731	0.712	0.704
	wikiSERA-5	0.839	0.884	**0.889**	0.856	**0.827**	0.878	**0.9**	0.861	0.631	0.708	**0.729**	0.675
	wikiSERA-10	**0.861**	**0.891**	0.885	0.865	0.822	**0.908**	0.889	0.868	**0.639**	0.731	0.712	0.685
	wikiSERA-DIS-5	0.825	0.879	0.878	0.856	0.817	0.87	0.873	0.86	0.624	0.701	0.691	0.678
	wikiSERA-DIS-10	0.848	**0.891**	0.885	**0.874**	0.815	**0.908**	0.874	**0.87**	0.626	**0.74**	0.693	**0.691**

Table A.40: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.82	0.824	0.874	0.833	0.783	0.816	0.875	0.827	0.6	0.643	0.689	0.633
	SERA-10	0.843	0.877	0.886	0.866	0.791	0.884	**0.908**	**0.881**	0.609	0.7	**0.726**	**0.702**
	SERA-NP-5	**0.848**	0.897	0.878	0.815	0.803	0.892	0.893	0.81	0.606	0.721	0.702	0.616
	SERA-NP-10	0.841	0.89	0.873	**0.869**	**0.819**	**0.907**	0.88	**0.881**	**0.629**	0.73	0.7	**0.702**
	SERA-KW-5	0.825	0.827	0.871	0.829	0.798	0.816	0.865	0.824	0.61	0.638	0.676	0.633
	SERA-KW-10	0.843	0.878	**0.889**	0.866	0.817	0.888	0.903	0.877	0.627	0.705	0.725	0.695
	SERA-DIS-5	0.834	0.829	0.862	0.833	0.782	0.808	0.857	0.829	0.593	0.63	0.671	0.641
	SERA-DIS-10	0.829	0.869	0.881	0.862	0.784	0.87	0.884	0.864	0.597	0.69	0.702	0.673
	SERA-DIS-NP-5	0.826	0.886	0.871	0.817	0.794	0.874	0.885	0.796	0.595	0.704	0.7	0.597
	SERA-DIS-NP-10	0.835	**0.899**	0.874	0.858	0.815	0.904	0.884	0.848	0.619	**0.734**	0.695	0.665
	SERA-DIS-KW-5	0.82	0.83	0.863	0.826	0.791	0.814	0.854	0.824	0.606	0.641	0.664	0.635
	SERA-DIS-KW-10	0.837	0.87	0.881	0.861	0.808	0.875	0.88	0.863	0.62	0.694	0.694	0.672
	wikiSERA-5	0.843	0.865	0.875	0.84	**0.84**	0.86	**0.885**	0.86	**0.655**	0.684	0.703	0.671
	wikiSERA-10	**0.858**	0.877	**0.879**	0.866	0.806	**0.897**	0.882	**0.864**	0.629	**0.72**	**0.705**	**0.689**
	wikiSERA-DIS-5	0.834	0.864	0.853	0.841	0.827	0.844	0.855	0.859	0.645	0.676	0.671	0.675
	wikiSERA-DIS-10	0.849	**0.878**	0.87	**0.868**	0.806	0.886	0.866	0.859	0.63	0.705	0.687	0.675

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.82	0.825	0.873	0.833	0.782	0.814	0.872	0.823	0.597	0.641	0.687	0.631
	SERA-10	0.842	0.876	0.886	0.866	0.789	0.883	**0.905**	**0.878**	0.609	0.698	**0.724**	**0.7**
	SERA-NP-5	**0.847**	0.897	0.878	0.815	0.801	0.891	0.892	0.809	0.603	0.718	0.7	0.614
	SERA-NP-10	0.84	0.89	0.872	**0.869**	**0.817**	**0.905**	0.877	**0.878**	0.626	0.728	0.698	**0.7**
	SERA-KW-5	0.825	0.827	0.871	0.828	0.796	0.813	0.863	0.82	0.61	0.636	0.673	0.63
	SERA-KW-10	0.842	0.878	**0.888**	0.866	0.81	0.887	0.9	0.874	**0.627**	0.702	0.723	0.693
	SERA-DIS-5	0.833	0.829	0.862	0.832	0.779	0.807	0.856	0.825	0.591	0.627	0.668	0.638
	SERA-DIS-10	0.828	0.869	0.881	0.862	0.781	0.868	0.882	0.861	0.597	0.687	0.7	0.671
	SERA-DIS-NP-5	0.825	0.886	0.871	0.817	0.791	0.873	0.884	0.794	0.592	0.701	0.698	0.595
	SERA-DIS-NP-10	0.834	**0.899**	0.874	0.858	0.812	0.903	0.883	0.846	0.616	**0.731**	0.693	0.662
	SERA-DIS-KW-5	0.819	0.83	0.863	0.825	0.788	0.812	0.852	0.82	0.603	0.638	0.661	0.633
	SERA-DIS-KW-10	0.839	0.87	0.881	0.86	0.806	0.873	0.878	0.86	0.614	0.691	0.691	0.67
	wikiSERA-5	0.842	0.865	0.875	0.839	**0.837**	0.857	**0.884**	0.857	**0.653**	0.682	0.7	0.668
	wikiSERA-10	**0.857**	0.877	**0.878**	0.866	0.803	**0.896**	0.881	**0.863**	0.626	**0.718**	**0.702**	**0.686**
	wikiSERA-DIS-5	0.833	0.864	0.853	0.841	0.823	0.842	0.854	0.857	0.643	0.673	0.668	0.672
	wikiSERA-DIS-10	0.848	**0.878**	0.869	**0.868**	0.803	0.885	0.865	0.857	0.627	0.702	0.684	0.672

Table A.41: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.826	0.831	0.88	0.841	0.774	0.839	0.885	0.85	0.592	0.673	0.707	0.664
	SERA-10	0.846	0.884	0.886	0.87	0.797	0.9	**0.912**	0.89	0.616	0.724	**0.739**	0.711
	SERA-NP-5	**0.853**	0.9	0.884	0.827	0.817	0.898	0.896	0.833	0.618	0.733	0.71	0.644
	SERA-NP-10	0.845	0.89	0.877	0.867	**0.822**	0.912	0.89	0.884	**0.637**	0.738	0.712	0.702
	SERA-KW-5	0.822	0.837	0.878	0.837	0.798	0.838	0.883	0.846	0.609	0.667	0.695	0.659
	SERA-KW-10	0.841	0.885	**0.89**	0.87	0.815	0.906	0.908	0.89	0.627	0.727	0.734	**0.713**
	SERA-DIS-5	0.835	0.833	0.871	0.846	0.783	0.825	0.87	0.856	0.598	0.655	0.693	0.672
	SERA-DIS-10	0.827	0.877	0.882	**0.871**	0.792	0.887	0.888	**0.891**	0.602	0.707	0.708	0.71
	SERA-DIS-NP-5	0.828	0.894	0.879	0.833	0.796	0.884	0.894	0.828	0.606	0.718	0.713	0.637
	SERA-DIS-NP-10	0.837	**0.902**	0.881	0.862	0.811	**0.917**	0.891	0.874	0.621	**0.754**	0.714	0.685
	SERA-DIS-KW-5	0.812	0.838	0.873	0.84	0.791	0.837	0.872	0.849	0.606	0.667	0.691	0.659
	SERA-DIS-KW-10	0.832	0.881	0.887	0.87	0.807	0.894	0.891	0.886	0.626	0.719	0.706	0.702
	wikiSERA-5	0.84	0.873	**0.885**	0.851	**0.829**	0.865	**0.897**	0.864	0.634	0.698	0.72	0.685
	wikiSERA-10	**0.858**	**0.883**	0.884	0.865	0.818	**0.903**	0.894	**0.877**	0.632	**0.727**	**0.727**	**0.698**
	wikiSERA-DIS-5	0.832	0.87	0.869	0.852	0.828	0.865	0.87	0.86	**0.638**	0.701	0.685	0.671
	wikiSERA-DIS-10	0.849	0.882	0.878	**0.87**	0.809	0.899	0.875	0.874	0.626	0.722	0.695	0.69

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.826	0.831	0.88	0.841	0.773	0.836	0.884	0.846	0.59	0.671	0.704	0.661
	SERA-10	0.845	0.884	0.885	0.87	0.795	0.899	**0.909**	**0.888**	0.616	0.722	**0.736**	0.708
	SERA-NP-5	**0.853**	0.9	0.883	0.827	0.815	0.897	0.895	0.831	0.616	0.73	0.708	0.641
	SERA-NP-10	0.845	0.889	0.876	0.866	**0.82**	0.911	0.888	0.881	**0.635**	0.735	0.709	0.7
	SERA-KW-5	0.822	0.837	0.878	0.837	0.796	0.836	0.882	0.843	0.609	0.664	0.692	0.656
	SERA-KW-10	0.842	0.885	**0.889**	0.87	0.811	0.905	0.906	0.887	0.624	0.725	0.732	**0.711**
	SERA-DIS-5	0.835	0.833	0.871	0.846	0.78	0.823	0.868	0.852	0.596	0.653	0.69	0.67
	SERA-DIS-10	0.826	0.877	0.882	**0.871**	0.789	0.885	0.886	**0.888**	0.6	0.705	0.706	0.707
	SERA-DIS-NP-5	0.827	0.894	0.878	0.833	0.794	0.883	0.893	0.826	0.603	0.716	0.711	0.635
	SERA-DIS-NP-10	0.836	**0.902**	0.881	0.862	0.808	**0.915**	0.889	0.871	0.619	**0.752**	0.712	0.683
	SERA-DIS-KW-5	0.811	0.838	0.873	0.84	0.788	0.835	0.871	0.845	0.604	0.665	0.689	0.656
	SERA-DIS-KW-10	0.83	0.88	0.886	0.869	0.803	0.892	0.889	0.883	0.616	0.717	0.704	0.7
	wikiSERA-5	0.839	0.873	**0.885**	0.851	**0.826**	0.862	**0.896**	0.862	0.631	0.695	0.718	0.683
	wikiSERA-10	**0.857**	**0.883**	0.883	0.864	0.816	**0.902**	0.892	**0.875**	0.63	**0.725**	**0.725**	**0.696**
	wikiSERA-DIS-5	0.831	0.87	0.868	0.852	0.825	0.862	0.869	0.857	**0.636**	0.699	0.683	0.668
	wikiSERA-DIS-10	0.848	0.882	0.878	**0.87**	0.806	0.898	0.873	0.872	0.624	0.719	0.692	0.688

Table A.42: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.6 Correlation of SERA and wikiSERA with Responsiveness on TAC2008/Wikipedia

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.718	0.742	0.786	0.709	0.669	0.742	0.781	0.726	0.474	0.545	0.593	0.538
SERA-10	0.724	0.812	0.785	0.758	0.683	0.811	0.803	0.789	0.49	0.616	0.613	0.592
SERA-NP-5	0.722	0.825	0.785	0.696	**0.709**	0.801	0.784	0.691	**0.517**	0.607	0.59	0.505
SERA-NP-10	**0.728**	0.767	0.774	0.725	0.659	0.768	0.772	0.751	0.483	0.576	0.58	0.556
SERA-KW-5	0.688	0.756	0.784	0.7	0.639	0.744	0.773	0.72	0.459	0.553	0.585	0.534
SERA-KW-10	0.705	0.808	**0.796**	0.765	0.651	0.803	**0.809**	0.8	0.471	0.609	**0.621**	0.608
SERA-DIS-5	0.69	0.749	0.783	0.733	0.642	0.754	0.786	0.774	0.462	0.55	0.592	0.58
SERA-DIS-10	0.677	0.816	0.775	**0.768**	0.629	**0.822**	0.776	**0.813**	0.444	**0.622**	0.579	0.608
SERA-DIS-NP-5	0.686	**0.828**	0.78	0.729	0.655	0.795	0.776	0.706	0.456	0.601	0.574	0.519
SERA-DIS-NP-10	0.705	0.796	0.788	0.749	0.631	0.799	0.785	0.761	0.459	0.609	0.592	0.563
SERA-DIS-KW-5	0.642	0.759	0.788	0.718	0.596	0.748	0.793	0.741	0.422	0.539	0.598	0.552
SERA-DIS-KW-10	0.657	0.822	0.79	**0.768**	0.615	**0.822**	0.791	0.81	0.44	0.62	0.594	**0.621**
wikiSERA-5	0.697	0.801	0.773	0.711	0.657	**0.811**	0.751	0.724	0.481	**0.62**	0.566	0.55
wikiSERA-10	**0.744**	**0.804**	**0.782**	0.721	0.674	0.802	**0.791**	0.752	0.486	0.606	**0.605**	0.556
wikiSERA-DIS-5	0.678	0.77	0.763	0.723	0.66	0.767	0.734	0.731	0.482	0.574	0.551	0.551
wikiSERA-DIS-10	0.723	0.798	0.778	**0.736**	**0.676**	0.792	0.766	**0.77**	**0.495**	0.602	0.587	**0.579**

Table A.43: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.729	0.775	0.795	0.724	0.674	0.745	0.745	0.686	0.484	0.556	0.56	0.51
SERA-10	0.738	0.823	0.805	0.746	0.655	0.794	0.799	0.736	0.461	0.604	0.609	0.536
SERA-NP-5	0.723	**0.856**	0.788	0.701	0.667	**0.848**	0.745	0.642	0.477	**0.658**	0.55	0.474
SERA-NP-10	0.748	0.799	0.776	**0.753**	**0.728**	0.759	0.732	**0.746**	**0.525**	0.563	0.535	0.535
SERA-KW-5	0.735	0.788	0.792	0.722	0.676	0.743	0.742	0.687	0.471	0.558	0.556	0.509
SERA-KW-10	**0.753**	0.818	**0.816**	0.746	0.692	0.788	**0.805**	0.743	0.485	0.605	**0.616**	**0.547**
SERA-DIS-5	0.724	0.795	0.775	0.741	0.667	0.763	0.708	0.674	0.472	0.569	0.514	0.501
SERA-DIS-10	0.703	0.825	0.795	0.747	0.618	0.791	0.755	0.717	0.423	0.598	0.562	0.519
SERA-DIS-NP-5	0.668	0.849	0.815	0.723	0.622	0.81	0.758	0.63	0.433	0.609	0.573	0.449
SERA-DIS-NP-10	0.716	0.826	0.808	0.749	0.692	0.792	0.763	0.695	0.494	0.588	0.571	0.502
SERA-DIS-KW-5	0.673	0.808	0.779	0.74	0.656	0.771	0.712	0.677	0.46	0.584	0.524	0.5
SERA-DIS-KW-10	0.693	0.82	0.805	0.749	0.669	0.787	0.765	0.711	0.48	0.597	0.573	0.518
wikiSERA-5	0.735	0.821	**0.791**	0.743	**0.72**	0.77	0.752	0.73	0.517	0.592	0.564	0.54
wikiSERA-10	**0.771**	0.828	0.786	0.768	0.715	0.809	**0.765**	**0.76**	**0.523**	0.624	**0.571**	**0.562**
wikiSERA-DIS-5	0.683	0.818	0.776	0.747	0.645	0.782	0.702	0.687	0.45	0.593	0.517	0.5
wikiSERA-DIS-10	0.734	**0.838**	0.784	**0.772**	0.677	**0.823**	0.749	0.744	0.482	**0.626**	0.553	0.548

Table A.44: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_2$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.734	0.667	0.752	0.707	0.66	0.664	0.742	0.699	0.477	0.471	0.535	0.501
SERA-10	0.749	0.766	0.751	0.76	0.68	0.751	**0.792**	**0.789**	0.484	0.558	**0.592**	**0.593**
SERA-NP-5	0.732	0.741	**0.764**	0.647	0.644	0.728	0.773	0.631	0.466	0.538	0.572	0.447
SERA-NP-10	0.74	0.785	0.731	0.733	0.686	**0.798**	0.745	0.771	0.487	**0.601**	0.544	0.56
SERA-KW-5	0.741	0.669	0.747	0.705	0.691	0.669	0.733	0.697	0.509	0.481	0.528	0.504
SERA-KW-10	**0.759**	0.771	0.753	**0.761**	**0.708**	0.761	0.785	0.787	**0.528**	0.563	0.586	0.586
SERA-DIS-5	0.721	0.686	0.715	0.678	0.639	0.668	0.705	0.685	0.448	0.462	0.507	0.495
SERA-DIS-10	0.739	0.749	0.728	0.728	0.687	0.727	0.745	0.75	0.493	0.524	0.547	0.551
SERA-DIS-NP-5	0.687	0.752	0.723	0.596	0.638	0.724	0.757	0.573	0.444	0.533	0.548	0.411
SERA-DIS-NP-10	0.698	**0.797**	0.72	0.676	0.667	0.797	0.736	0.703	0.474	0.592	0.533	0.503
SERA-DIS-KW-5	0.73	0.677	0.71	0.667	0.687	0.639	0.682	0.665	0.49	0.44	0.488	0.482
SERA-DIS-KW-10	0.749	0.751	0.726	0.716	0.703	0.732	0.737	0.742	0.503	0.527	0.544	0.541
wikiSERA-5	**0.747**	0.729	0.756	0.684	**0.722**	0.727	0.734	0.718	**0.536**	0.539	0.544	0.516
wikiSERA-10	0.731	**0.762**	**0.762**	**0.739**	0.631	**0.75**	**0.777**	**0.76**	0.445	**0.553**	**0.58**	**0.567**
wikiSERA-DIS-5	0.733	0.728	0.694	0.682	0.717	0.695	0.674	0.711	0.528	0.501	0.491	0.512
wikiSERA-DIS-10	0.728	0.74	0.726	0.728	0.662	0.719	0.727	0.741	0.476	0.524	0.543	0.544

Table A.45: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_3$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.716	0.75	0.796	0.692	0.641	0.719	0.769	0.695	0.473	0.527	0.588	0.507
SERA-10	0.745	0.778	0.8	0.774	0.666	0.759	0.812	0.774	0.468	0.572	0.62	0.584
SERA-NP-5	0.763	**0.808**	0.769	0.708	0.669	0.778	0.776	0.699	0.475	0.59	0.577	0.527
SERA-NP-10	0.715	0.788	0.762	0.754	0.649	0.793	0.765	0.739	0.46	0.593	0.573	0.53
SERA-KW-5	0.751	0.737	0.792	0.68	0.703	0.719	0.78	0.679	0.514	0.527	0.596	0.49
SERA-KW-10	**0.769**	0.782	0.8	**0.776**	**0.72**	0.767	0.798	**0.785**	**0.534**	0.583	0.606	**0.587**
SERA-DIS-5	0.726	0.785	0.799	0.704	0.62	0.728	0.788	0.701	0.445	0.538	0.603	0.517
SERA-DIS-10	0.721	0.799	**0.816**	**0.776**	0.656	0.755	**0.824**	0.77	0.463	0.565	**0.631**	0.577
SERA-DIS-NP-5	0.762	0.793	0.758	0.725	0.693	0.735	0.778	0.7	0.493	0.551	0.57	0.516
SERA-DIS-NP-10	0.736	0.801	0.756	0.764	0.666	**0.799**	0.775	0.737	0.471	**0.596**	0.573	0.536
SERA-DIS-KW-5	0.75	0.77	0.803	0.693	0.697	0.721	0.796	0.696	0.51	0.528	0.61	0.502
SERA-DIS-KW-10	0.765	0.798	0.811	0.774	0.714	0.762	0.806	0.757	0.528	0.567	0.612	0.563
wikiSERA-5	**0.773**	0.767	0.775	0.711	**0.72**	0.731	**0.779**	0.705	**0.517**	0.537	**0.577**	0.516
wikiSERA-10	0.752	0.774	**0.782**	0.761	0.67	0.764	0.761	**0.757**	0.48	0.558	0.573	**0.565**
wikiSERA-DIS-5	0.756	0.784	0.758	0.716	0.707	0.744	0.756	0.689	0.516	0.55	0.557	0.5
wikiSERA-DIS-10	0.757	**0.789**	0.776	**0.762**	0.647	**0.775**	0.748	0.749	0.462	**0.575**	0.561	0.557

Table A.46: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_4$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.743	0.754	0.801	0.74	0.695	0.749	0.789	0.72	0.511	0.552	0.593	0.533
SERA-10	0.752	0.817	0.795	0.771	0.683	0.812	0.815	**0.793**	0.489	0.612	0.625	**0.594**
SERA-NP-5	0.743	**0.84**	**0.81**	0.713	0.685	0.816	0.793	0.696	0.496	0.61	0.599	0.516
SERA-NP-10	0.756	0.801	0.777	0.757	**0.713**	0.793	0.773	0.769	0.511	0.594	0.575	0.557
SERA-KW-5	0.74	0.765	0.797	0.737	0.691	0.761	0.782	0.728	0.499	0.563	0.581	0.544
SERA-KW-10	**0.758**	0.816	0.804	**0.773**	0.711	0.818	**0.818**	0.788	**0.514**	**0.622**	**0.63**	0.59
SERA-DIS-5	0.734	0.77	0.784	0.748	0.677	0.752	0.766	0.743	0.485	0.558	0.577	0.547
SERA-DIS-10	0.721	0.814	0.784	0.771	0.661	0.81	0.772	0.791	0.471	0.607	0.58	0.587
SERA-DIS-NP-5	0.698	0.839	0.802	0.714	0.653	**0.819**	0.781	0.692	0.461	0.613	0.584	0.511
SERA-DIS-NP-10	0.725	0.823	0.793	0.744	0.684	0.815	0.778	0.75	0.485	0.62	0.585	0.547
SERA-DIS-KW-5	0.703	0.776	0.786	0.741	0.676	0.759	0.771	0.734	0.479	0.565	0.58	0.547
SERA-DIS-KW-10	0.72	0.816	0.794	0.767	0.691	0.808	0.783	0.775	0.494	0.608	0.59	0.576
wikiSERA-5	0.746	0.806	**0.8**	0.745	**0.715**	0.786	0.776	0.748	**0.516**	0.59	0.592	0.563
wikiSERA-10	**0.762**	**0.813**	0.792	0.76	0.678	**0.815**	**0.802**	**0.783**	0.487	**0.611**	**0.601**	**0.579**
wikiSERA-DIS-5	0.722	0.798	0.777	0.747	0.693	0.775	0.732	0.746	0.499	0.575	0.54	0.552
wikiSERA-DIS-10	0.743	0.809	0.782	**0.764**	0.68	0.796	0.765	0.775	0.487	0.59	0.573	0.573

Table A.47: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.738	0.781	0.816	0.738	0.67	0.755	0.8	0.716	0.491	0.564	0.609	0.528
SERA-10	0.75	0.821	0.81	0.777	0.679	0.799	0.819	0.793	0.48	0.596	**0.625**	0.592
SERA-NP-5	0.756	**0.852**	0.807	0.732	0.7	**0.823**	0.8	0.71	0.503	**0.627**	0.605	0.528
SERA-NP-10	0.746	0.805	0.788	0.764	0.711	0.803	0.774	0.764	0.505	0.602	0.574	0.553
SERA-KW-5	0.744	0.787	0.812	0.729	0.703	0.764	0.805	0.721	0.51	0.565	0.613	0.531
SERA-KW-10	**0.763**	0.819	0.818	0.781	**0.723**	0.81	**0.822**	0.788	**0.521**	0.613	0.624	0.593
SERA-DIS-5	0.734	0.799	0.811	0.763	0.657	0.773	0.785	0.753	0.474	0.579	0.601	0.558
SERA-DIS-10	0.715	0.83	0.812	0.787	0.643	0.809	0.794	**0.801**	0.45	0.608	0.601	**0.596**
SERA-DIS-NP-5	0.728	0.849	0.812	0.759	0.676	0.819	0.802	0.717	0.479	0.626	0.608	0.538
SERA-DIS-NP-10	0.738	0.826	0.806	0.773	0.687	0.815	0.796	0.767	0.49	0.615	0.597	0.562
SERA-DIS-KW-5	0.708	0.804	0.816	0.755	0.665	0.769	0.798	0.734	0.468	0.58	0.608	0.546
SERA-DIS-KW-10	0.726	0.83	**0.821**	**0.789**	0.683	0.812	0.808	0.8	0.483	0.615	0.614	0.594
wikiSERA-5	0.755	0.816	**0.806**	0.751	**0.725**	0.791	0.797	0.742	**0.524**	0.598	0.603	0.561
wikiSERA-10	**0.772**	0.817	0.798	0.767	0.706	0.815	**0.801**	**0.778**	0.514	0.61	**0.607**	**0.581**
wikiSERA-DIS-5	0.726	0.814	0.797	0.76	0.695	0.779	0.765	0.741	0.504	0.586	0.57	0.557
wikiSERA-DIS-10	0.752	**0.825**	0.799	**0.777**	0.683	**0.817**	0.778	0.773	0.487	**0.624**	0.584	0.574

Table A.48: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.74	0.755	0.802	0.727	0.677	0.723	0.783	0.699	0.495	0.525	0.59	0.503
SERA-10	0.754	0.803	0.8	0.774	0.675	0.781	**0.817**	**0.783**	0.478	0.583	**0.62**	**0.582**
SERA-NP-5	0.754	**0.835**	0.804	0.715	0.68	0.808	0.796	0.69	0.477	0.604	0.592	0.505
SERA-NP-10	0.746	0.808	0.775	0.765	0.691	0.807	0.761	0.758	0.489	0.609	0.559	0.548
SERA-KW-5	0.756	0.754	0.796	0.722	0.707	0.722	0.775	0.697	0.508	0.525	0.575	0.506
SERA-KW-10	**0.775**	0.803	**0.805**	**0.775**	**0.72**	0.781	0.811	0.781	**0.519**	0.585	0.618	**0.582**
SERA-DIS-5	0.741	0.781	0.79	0.732	0.653	0.743	0.762	0.708	0.463	0.548	0.57	0.52
SERA-DIS-10	0.732	0.807	0.795	0.766	0.662	0.774	0.792	0.762	0.467	0.576	0.594	0.561
SERA-DIS-NP-5	0.726	0.831	0.798	0.715	0.665	0.794	0.788	0.663	0.467	0.596	0.588	0.484
SERA-DIS-NP-10	0.731	0.826	0.786	0.752	0.685	**0.813**	0.775	0.719	0.476	**0.618**	0.575	0.522
SERA-DIS-KW-5	0.735	0.778	0.79	0.723	0.689	0.744	0.763	0.701	0.495	0.552	0.563	0.515
SERA-DIS-KW-10	0.751	0.806	0.798	0.764	0.708	0.779	0.788	0.76	0.508	0.582	0.591	0.554
wikiSERA-5	**0.768**	0.789	**0.791**	0.732	**0.743**	0.755	0.781	0.73	**0.536**	0.554	0.581	0.533
wikiSERA-10	0.763	0.801	0.789	**0.77**	0.676	**0.795**	**0.791**	**0.772**	0.485	**0.592**	**0.599**	**0.575**
wikiSERA-DIS-5	0.743	0.797	0.766	0.737	0.709	0.754	0.74	0.727	0.517	0.567	0.545	0.536
wikiSERA-DIS-10	0.751	**0.806**	0.777	0.768	0.673	0.785	0.758	0.757	0.491	0.579	0.563	0.562

Table A.49: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.742	0.763	**0.809**	0.736	0.669	0.751	0.796	0.724	0.488	0.56	0.605	0.53
SERA-10	0.754	0.812	0.801	0.777	0.685	0.798	**0.823**	0.793	0.486	0.594	**0.627**	**0.594**
SERA-NP-5	0.756	**0.839**	**0.809**	0.722	0.691	0.819	0.8	0.713	0.489	0.616	0.603	0.533
SERA-NP-10	0.75	0.806	0.78	0.763	0.703	0.812	0.772	0.771	0.501	0.618	0.571	0.561
SERA-KW-5	0.749	0.767	0.804	0.729	0.708	0.749	0.792	0.717	0.514	0.552	0.599	0.526
SERA-KW-10	**0.764**	0.812	0.808	**0.78**	**0.725**	0.806	0.818	0.793	**0.526**	0.603	0.623	**0.594**
SERA-DIS-5	0.739	0.781	0.799	0.748	0.652	0.757	0.778	0.738	0.466	0.568	0.59	0.546
SERA-DIS-10	0.726	0.815	0.798	0.778	0.664	0.79	0.797	**0.796**	0.472	0.588	0.603	0.591
SERA-DIS-NP-5	0.725	0.838	0.803	0.73	0.671	0.809	0.798	0.7	0.478	0.604	0.602	0.52
SERA-DIS-NP-10	0.734	0.826	0.793	0.757	0.681	**0.82**	0.787	0.75	0.482	**0.626**	0.586	0.55
SERA-DIS-KW-5	0.722	0.783	0.801	0.739	0.681	0.761	0.786	0.727	0.484	0.567	0.591	0.537
SERA-DIS-KW-10	0.742	0.817	0.805	0.776	0.694	0.797	0.802	0.789	0.497	0.598	0.601	0.584
wikiSERA-5	0.759	0.803	**0.801**	0.741	**0.725**	0.774	0.784	0.739	**0.527**	0.581	0.589	0.55
wikiSERA-10	**0.765**	0.808	0.794	0.765	0.69	**0.805**	**0.806**	**0.79**	0.501	0.598	**0.606**	**0.587**
wikiSERA-DIS-5	0.736	0.802	0.782	0.746	0.707	0.773	0.749	0.73	0.513	0.58	0.55	0.54
wikiSERA-DIS-10	0.751	**0.81**	0.786	**0.769**	0.673	0.8	0.774	0.769	0.479	**0.601**	0.579	0.569

Table A.50: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2008/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.7 Correlation of SERA and wikiSERA with Pyramid on TAC2009/Wikipedia

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.816	0.881	0.928	0.921	0.788	0.844	0.84	0.885	0.619	0.689	0.674	0.724
	SERA-10	0.856	0.899	0.91	0.924	0.85	0.841	**0.855**	**0.924**	0.679	0.674	**0.693**	**0.777**
	SERA-NP-5	0.808	0.881	0.897	0.899	0.737	0.789	0.753	0.868	0.576	0.615	0.601	0.716
	SERA-NP-10	0.844	0.896	0.897	0.909	0.847	**0.859**	0.831	0.856	0.673	**0.713**	0.683	0.694
	SERA-KW-5	0.815	0.876	0.92	0.914	0.789	0.844	0.811	0.861	0.615	0.684	0.64	0.699
	SERA-KW-10	0.853	0.901	0.9	0.925	**0.855**	0.833	0.848	0.922	**0.688**	0.666	0.69	0.77
	SERA-DIS-5	0.86	0.908	0.939	0.95	0.787	0.828	0.829	0.881	0.612	0.663	0.665	0.719
	SERA-DIS-10	**0.912**	0.939	**0.946**	**0.964**	0.842	0.822	0.834	0.915	0.667	0.657	0.675	0.774
	SERA-DIS-NP-5	0.858	0.922	0.926	0.938	0.742	0.818	0.752	0.862	0.574	0.643	0.596	0.702
	SERA-DIS-NP-10	**0.912**	**0.942**	0.937	0.95	0.83	0.853	0.802	0.844	0.659	0.697	0.636	0.681
	SERA-DIS-KW-5	0.857	0.902	0.932	0.944	0.788	0.823	0.82	0.855	0.617	0.663	0.651	0.683
	SERA-DIS-KW-10	0.906	0.934	0.941	0.959	0.83	0.814	0.831	0.907	0.659	0.64	0.671	0.754
	wikiSERA-5	0.82	0.904	0.907	0.905	0.808	0.848	0.795	0.864	0.642	0.69	0.633	0.704
	wikiSERA-10	0.849	0.923	0.899	0.911	**0.859**	0.857	**0.853**	**0.899**	**0.693**	0.702	**0.684**	**0.753**
	wikiSERA-DIS-5	0.867	0.922	0.931	0.941	0.787	0.83	0.794	0.859	0.618	0.685	0.638	0.696
	wikiSERA-DIS-10	**0.913**	**0.953**	**0.945**	**0.961**	0.849	**0.882**	0.829	0.892	0.677	**0.731**	0.662	0.739

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.815	0.88	0.928	0.921	0.791	0.846	0.84	0.885	0.624	0.691	0.672	0.723
	SERA-10	0.856	0.898	0.91	0.923	0.853	0.842	**0.856**	**0.926**	0.684	0.673	**0.697**	**0.782**
	SERA-NP-5	0.807	0.881	0.897	0.899	0.739	0.789	0.759	0.869	0.58	0.615	0.606	0.719
	SERA-NP-10	0.843	0.896	0.897	0.909	0.848	**0.861**	0.833	0.858	0.675	**0.711**	0.685	0.698
	SERA-KW-5	0.815	0.875	0.919	0.914	0.792	0.847	0.812	0.861	0.62	0.686	0.642	0.699
	SERA-KW-10	0.852	0.9	0.9	0.925	**0.857**	0.834	0.849	0.925	**0.693**	0.665	0.694	0.775
	SERA-DIS-5	0.86	0.908	0.939	0.95	0.79	0.83	0.83	0.882	0.617	0.666	0.667	0.721
	SERA-DIS-10	**0.912**	0.939	**0.946**	**0.964**	0.844	0.821	0.836	0.917	0.672	0.655	0.679	0.776
	SERA-DIS-NP-5	0.858	0.922	0.926	0.938	0.744	0.818	0.755	0.863	0.577	0.641	0.6	0.703
	SERA-DIS-NP-10	**0.912**	**0.942**	0.937	0.95	0.832	0.854	0.804	0.846	0.662	0.697	0.637	0.682
	SERA-DIS-KW-5	0.856	0.902	0.932	0.944	0.79	0.825	0.822	0.855	0.622	0.665	0.656	0.685
	SERA-DIS-KW-10	0.905	0.933	0.941	0.959	0.832	0.814	0.833	0.91	0.664	0.639	0.675	0.757
	wikiSERA-5	0.82	0.904	0.907	0.904	0.811	0.85	0.796	0.865	0.647	0.694	0.634	0.705
	wikiSERA-10	0.849	0.923	0.899	0.911	**0.861**	0.858	**0.855**	**0.901**	**0.698**	0.701	**0.689**	**0.755**
	wikiSERA-DIS-5	0.866	0.922	0.931	0.941	0.788	0.831	0.796	0.86	0.62	0.683	0.64	0.698
	wikiSERA-DIS-10	**0.913**	**0.952**	**0.945**	**0.961**	0.85	**0.882**	0.831	0.894	0.682	**0.73**	0.664	0.741

Table A.51: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.869	0.927	0.903	0.919	0.807	0.818	0.758	0.766	0.643	0.648	0.6	0.604
	SERA-10	0.909	0.926	0.926	0.935	0.872	0.839	0.791	0.861	0.711	0.668	0.624	0.71
	SERA-NP-5	0.875	0.908	0.895	0.91	0.803	0.741	0.738	0.792	0.631	0.563	0.57	0.616
	SERA-NP-10	0.895	0.931	**0.942**	0.939	0.865	0.829	**0.878**	0.86	0.697	0.671	**0.72**	0.702
	SERA-KW-5	0.869	0.925	0.894	0.919	0.808	0.81	0.763	0.767	0.646	0.637	0.609	0.599
	SERA-KW-10	0.913	0.92	0.92	0.933	0.877	0.827	0.796	0.865	0.713	0.667	0.621	0.705
	SERA-DIS-5	0.914	**0.944**	0.91	0.925	0.813	0.802	0.729	0.756	0.635	0.62	0.573	0.587
	SERA-DIS-10	0.949	0.943	0.922	**0.946**	0.877	0.837	0.777	**0.877**	0.725	0.67	0.605	**0.728**
	SERA-DIS-NP-5	0.914	0.917	0.906	0.923	0.795	0.742	0.686	0.758	0.623	0.578	0.521	0.585
	SERA-DIS-NP-10	0.937	0.933	0.923	0.938	0.846	**0.852**	0.808	0.827	0.673	**0.688**	0.649	0.657
	SERA-DIS-KW-5	0.909	0.939	0.902	0.924	0.815	0.796	0.72	0.761	0.639	0.615	0.561	0.6
	SERA-DIS-KW-10	**0.951**	0.936	0.916	0.943	**0.885**	0.813	0.762	0.85	**0.729**	0.654	0.595	0.696
	wikiSERA-5	0.887	0.905	0.885	0.911	0.811	0.78	0.685	0.769	0.643	0.612	0.527	0.606
	wikiSERA-10	0.905	0.924	0.926	0.926	**0.879**	**0.83**	**0.817**	0.823	**0.72**	0.657	**0.663**	0.648
	wikiSERA-DIS-5	0.917	0.929	0.905	0.935	0.82	0.779	0.691	0.81	0.654	0.613	0.535	0.661
	wikiSERA-DIS-10	**0.947**	**0.946**	**0.934**	**0.948**	0.866	0.828	0.792	**0.851**	0.693	**0.673**	0.64	**0.704**
		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.869	0.927	0.904	0.919	0.81	0.819	0.761	0.769	0.645	0.651	0.605	0.606
	SERA-10	0.908	0.926	0.927	0.935	0.872	0.842	0.797	0.865	0.713	0.673	0.629	0.715
	SERA-NP-5	0.875	0.909	0.895	0.91	0.803	0.743	0.742	0.793	0.63	0.567	0.574	0.619
	SERA-NP-10	0.894	0.931	**0.942**	0.939	0.866	0.83	**0.88**	0.863	0.699	0.673	**0.722**	0.707
	SERA-KW-5	0.869	0.925	0.895	0.919	0.812	0.811	0.766	0.77	0.648	0.639	0.614	0.601
	SERA-KW-10	0.913	0.92	0.92	0.934	0.877	0.831	0.802	0.869	0.716	0.67	0.626	0.71
	SERA-DIS-5	0.913	**0.944**	0.91	0.925	0.815	0.803	0.731	0.76	0.637	0.622	0.575	0.59
	SERA-DIS-10	0.949	0.943	0.923	**0.947**	0.877	0.841	0.781	**0.882**	0.728	0.672	0.61	**0.732**
	SERA-DIS-NP-5	0.914	0.917	0.906	0.923	0.796	0.747	0.69	0.759	0.622	0.583	0.526	0.587
	SERA-DIS-NP-10	0.937	0.933	0.924	0.938	0.848	**0.855**	0.81	0.83	0.675	**0.693**	0.652	0.662
	SERA-DIS-KW-5	0.909	0.939	0.902	0.924	0.818	0.797	0.721	0.765	0.641	0.617	0.563	0.602
	SERA-DIS-KW-10	**0.951**	0.936	0.917	0.944	**0.885**	0.816	0.766	0.855	**0.732**	0.656	0.6	0.701
	wikiSERA-5	0.887	0.905	0.886	0.911	0.812	0.781	0.69	0.769	0.645	0.615	0.53	0.606
	wikiSERA-10	0.905	0.924	0.926	0.926	**0.881**	**0.831**	**0.819**	0.825	**0.722**	0.657	**0.662**	0.647
	wikiSERA-DIS-5	0.917	0.929	0.905	0.936	0.822	0.779	0.693	0.809	0.656	0.613	0.538	0.663
	wikiSERA-DIS-10	**0.947**	**0.946**	**0.934**	**0.948**	0.867	0.828	0.793	**0.854**	0.695	**0.672**	0.643	**0.706**

Table A.52: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_2$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.87	0.908	0.922	0.88	0.823	**0.872**	0.759	0.695	0.676	**0.731**	0.603	0.537
	SERA-10	0.88	0.919	0.923	0.931	**0.835**	0.804	0.802	**0.852**	**0.678**	0.65	0.637	**0.688**
	SERA-NP-5	0.864	0.873	0.902	0.875	0.804	0.753	0.763	0.757	0.645	0.594	0.588	0.58
	SERA-NP-10	0.873	0.91	0.917	0.902	0.773	0.807	0.809	0.783	0.62	0.65	0.623	0.608
	SERA-KW-5	0.868	0.904	0.913	0.878	0.811	0.855	0.738	0.691	0.658	0.715	0.576	0.524
	SERA-KW-10	0.884	0.915	0.926	0.93	0.826	0.791	0.816	0.845	0.666	0.636	0.657	0.668
	SERA-DIS-5	0.914	**0.942**	0.922	0.899	0.815	0.855	0.753	0.729	0.655	0.714	0.591	0.566
	SERA-DIS-10	0.936	0.938	0.93	**0.934**	0.825	0.816	0.809	0.817	0.665	0.653	0.661	0.657
	SERA-DIS-NP-5	0.902	0.911	0.922	0.906	0.825	0.761	0.761	0.728	0.657	0.605	0.582	0.557
	SERA-DIS-NP-10	0.932	0.931	0.922	0.922	0.811	0.783	0.796	0.738	0.646	0.613	0.621	0.566
	SERA-DIS-KW-5	0.908	0.94	0.919	0.894	0.819	0.847	0.744	0.738	0.659	0.705	0.591	0.566
	SERA-DIS-KW-10	**0.937**	0.935	**0.931**	**0.934**	0.826	0.817	**0.822**	0.833	0.665	0.655	**0.669**	0.678
	wikiSERA-5	0.851	0.912	0.91	0.897	0.818	**0.842**	0.766	0.745	**0.673**	**0.69**	0.602	0.575
	wikiSERA-10	0.879	0.926	0.925	0.91	0.823	0.841	0.784	0.794	0.669	0.685	0.614	0.62
	wikiSERA-DIS-5	0.897	0.937	0.918	0.924	0.801	0.826	0.769	0.78	0.644	0.682	0.602	0.608
	wikiSERA-DIS-10	**0.934**	**0.943**	**0.932**	**0.938**	**0.826**	0.826	**0.806**	**0.821**	0.663	0.681	**0.657**	**0.654**

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.869	0.908	0.922	0.88	0.824	**0.873**	0.76	0.696	0.677	**0.733**	0.607	0.537
	SERA-10	0.879	0.919	0.922	0.931	**0.835**	0.805	0.803	**0.854**	**0.678**	0.652	0.636	**0.688**
	SERA-NP-5	0.863	0.872	0.901	0.875	0.803	0.752	0.762	0.756	0.644	0.592	0.588	0.578
	SERA-NP-10	0.873	0.909	0.917	0.902	0.773	0.808	0.809	0.784	0.619	0.65	0.625	0.611
	SERA-KW-5	0.868	0.904	0.912	0.878	0.813	0.855	0.74	0.693	0.657	0.717	0.581	0.525
	SERA-KW-10	0.883	0.915	0.926	0.93	0.825	0.792	0.817	0.847	0.665	0.638	0.656	0.671
	SERA-DIS-5	0.914	**0.941**	0.922	0.899	0.816	0.855	0.754	0.731	0.657	0.712	0.594	0.569
	SERA-DIS-10	**0.936**	0.938	0.93	**0.934**	0.825	0.817	0.81	0.819	0.664	0.655	0.66	0.656
	SERA-DIS-NP-5	0.902	0.911	0.922	0.906	0.825	0.76	0.759	0.726	0.656	0.604	0.582	0.555
	SERA-DIS-NP-10	0.931	0.931	0.922	0.922	0.811	0.783	0.796	0.738	0.645	0.613	0.62	0.568
	SERA-DIS-KW-5	0.908	0.94	0.918	0.895	0.821	0.847	0.746	0.74	0.662	0.706	0.594	0.569
	SERA-DIS-KW-10	**0.936**	0.935	**0.931**	**0.934**	0.826	0.818	**0.824**	0.836	0.664	0.655	**0.668**	0.678
	wikiSERA-5	0.85	0.912	0.91	0.897	0.818	**0.843**	0.766	0.745	**0.673**	**0.691**	0.604	0.574
	wikiSERA-10	0.878	0.926	0.925	0.91	0.823	0.842	0.786	0.795	0.669	0.687	0.614	0.619
	wikiSERA-DIS-5	0.897	0.937	0.918	0.924	0.801	0.828	0.77	0.779	0.644	0.681	0.604	0.608
	wikiSERA-DIS-10	**0.934**	**0.943**	**0.932**	**0.938**	**0.827**	0.828	**0.807**	**0.821**	0.663	0.68	**0.657**	**0.653**

Table A.53: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_3$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.896	0.92	0.938	0.948	0.797	0.858	0.847	0.877	0.64	0.707	0.687	0.707
	SERA-10	0.926	0.946	0.943	0.948	0.852	0.878	0.838	0.866	0.695	**0.725**	0.674	0.708
	SERA-NP-5	0.921	0.924	0.927	0.935	0.836	0.852	0.766	0.844	0.675	0.692	0.614	0.684
	SERA-NP-10	0.927	0.933	**0.954**	0.943	0.863	0.838	0.865	0.806	0.7	0.672	0.714	0.65
	SERA-KW-5	0.9	0.914	0.939	0.941	0.803	0.856	0.83	**0.878**	0.648	0.697	0.666	0.707
	SERA-KW-10	0.925	**0.947**	0.94	0.948	0.831	**0.879**	0.849	0.863	0.668	0.724	0.683	0.704
	SERA-DIS-5	0.931	0.936	0.939	**0.951**	0.82	0.845	0.863	**0.878**	0.658	0.682	0.693	0.717
	SERA-DIS-10	**0.951**	0.944	0.934	0.946	0.873	0.875	0.88	0.876	**0.723**	0.721	**0.732**	**0.719**
	SERA-DIS-NP-5	0.939	0.941	0.94	0.944	0.85	0.846	0.753	0.821	0.677	0.673	0.596	0.655
	SERA-DIS-NP-10	0.946	0.942	0.933	0.934	**0.874**	0.855	0.806	0.797	0.714	0.697	0.647	0.635
	SERA-DIS-KW-5	0.926	0.931	0.935	0.944	0.825	0.831	0.853	0.868	0.662	0.665	0.691	0.693
	SERA-DIS-KW-10	0.943	0.94	0.933	0.941	0.862	0.875	**0.886**	0.858	0.705	0.715	**0.732**	0.696
	wikiSERA-5	0.89	0.911	0.924	0.923	0.785	0.822	0.799	0.816	0.628	0.661	0.638	0.653
	wikiSERA-10	0.93	0.934	**0.945**	0.944	**0.884**	**0.841**	0.863	**0.855**	**0.734**	**0.68**	0.704	**0.698**
	wikiSERA-DIS-5	0.928	0.928	0.938	0.944	0.804	0.809	0.822	0.842	0.647	0.636	0.657	0.675
	wikiSERA-DIS-10	**0.947**	**0.941**	0.941	**0.946**	0.863	0.832	**0.875**	0.848	0.715	0.662	**0.719**	0.694

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.895	0.92	0.938	0.948	0.801	0.858	0.849	0.877	0.645	0.705	0.689	0.706
	SERA-10	0.926	0.945	0.943	0.947	0.854	0.879	0.839	0.868	0.7	**0.73**	0.676	0.708
	SERA-NP-5	0.921	0.924	0.927	0.935	0.839	0.852	0.77	0.845	0.677	0.696	0.619	0.682
	SERA-NP-10	0.927	0.933	**0.954**	0.942	0.865	0.842	0.868	0.81	0.705	0.675	0.716	0.652
	SERA-KW-5	0.9	0.913	0.939	0.94	0.806	0.856	0.832	0.878	0.65	0.696	0.668	0.709
	SERA-KW-10	0.924	**0.947**	0.94	0.948	0.834	**0.881**	0.852	0.865	0.673	0.729	0.688	0.707
	SERA-DIS-5	0.931	0.936	0.939	**0.951**	0.823	0.845	0.866	**0.88**	0.66	0.68	0.698	0.719
	SERA-DIS-10	**0.951**	0.944	0.934	0.946	0.875	0.877	0.884	0.878	**0.728**	0.723	**0.737**	**0.723**
	SERA-DIS-NP-5	0.939	0.941	0.94	0.944	0.852	0.846	0.758	0.822	0.679	0.672	0.601	0.655
	SERA-DIS-NP-10	0.946	0.942	0.934	0.934	**0.876**	0.857	0.81	0.8	0.719	0.699	0.652	0.637
	SERA-DIS-KW-5	0.926	0.93	0.935	0.944	0.827	0.832	0.856	0.869	0.664	0.664	0.696	0.698
	SERA-DIS-KW-10	0.943	0.94	0.934	0.942	0.864	0.876	**0.889**	0.861	0.71	0.718	**0.737**	0.701
	wikiSERA-5	0.89	0.911	0.924	0.923	0.788	0.824	0.801	0.818	0.63	0.666	0.641	0.654
	wikiSERA-10	0.929	0.934	**0.944**	0.944	**0.885**	**0.844**	0.865	**0.856**	**0.737**	**0.683**	0.705	**0.697**
	wikiSERA-DIS-5	0.928	0.928	0.938	0.944	0.806	0.81	0.824	0.845	0.649	0.635	0.659	0.678
	wikiSERA-DIS-10	**0.947**	**0.941**	0.941	**0.946**	0.866	0.835	**0.876**	0.849	0.72	0.664	**0.718**	0.694

Table A.54: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_4$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.861	0.922	0.943	0.929	0.821	**0.88**	0.861	0.836	0.662	**0.734**	0.71	0.677
	SERA-10	0.89	0.927	0.933	0.938	0.863	0.873	0.851	0.896	0.717	0.726	0.7	0.735
	SERA-NP-5	0.861	0.906	0.921	0.917	0.812	0.787	0.802	0.857	0.64	0.617	0.632	0.697
	SERA-NP-10	0.881	0.926	0.934	0.928	0.846	0.868	**0.882**	0.848	0.686	0.716	0.719	0.679
	SERA-KW-5	0.86	0.92	0.938	0.928	0.823	0.868	0.843	0.825	0.661	0.724	0.685	0.656
	SERA-KW-10	0.893	0.926	0.93	0.938	0.864	0.861	0.853	0.892	0.714	0.708	0.697	0.735
	SERA-DIS-5	0.907	0.946	**0.946**	0.944	0.822	0.861	0.833	0.844	0.66	0.713	0.681	0.687
	SERA-DIS-10	**0.942**	**0.95**	0.943	**0.955**	**0.872**	0.866	0.856	0.899	**0.719**	0.713	0.719	**0.755**
	SERA-DIS-NP-5	0.903	0.936	0.938	0.94	0.813	0.828	0.788	0.837	0.646	0.662	0.624	0.674
	SERA-DIS-NP-10	0.936	0.947	0.939	0.944	0.847	0.856	0.844	0.841	0.682	0.708	0.688	0.678
	SERA-DIS-KW-5	0.902	0.943	0.941	0.942	0.829	0.853	0.82	0.845	0.663	0.698	0.658	0.685
	SERA-DIS-KW-10	0.941	0.945	0.939	0.952	0.867	0.851	0.863	**0.903**	0.713	0.698	**0.727**	0.754
	wikiSERA-5	0.862	0.923	0.921	0.921	0.823	0.87	0.801	0.839	0.664	0.73	0.646	0.681
	wikiSERA-10	0.886	0.937	0.931	0.927	**0.875**	**0.888**	**0.86**	0.866	**0.725**	**0.756**	**0.713**	0.705
	wikiSERA-DIS-5	0.904	0.944	0.935	0.949	0.828	0.855	0.787	0.852	0.665	0.722	0.63	0.706
	wikiSERA-DIS-10	**0.94**	**0.955**	**0.945**	**0.957**	0.871	0.88	0.843	**0.885**	0.713	0.745	0.698	**0.733**

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.861	0.922	0.944	0.929	0.824	**0.881**	0.864	0.838	0.666	**0.737**	0.715	0.679
	SERA-10	0.889	0.927	0.933	0.938	0.864	0.875	0.854	0.899	0.717	0.726	0.705	0.74
	SERA-NP-5	0.86	0.906	0.921	0.917	0.812	0.787	0.805	0.858	0.639	0.615	0.637	0.7
	SERA-NP-10	0.881	0.926	0.934	0.928	0.848	0.869	**0.883**	0.851	0.688	0.715	0.721	0.682
	SERA-KW-5	0.86	0.92	0.938	0.928	0.826	0.87	0.845	0.827	0.666	0.726	0.69	0.658
	SERA-KW-10	0.892	0.925	0.93	0.938	0.865	0.863	0.856	0.896	0.714	0.707	0.701	0.74
	SERA-DIS-5	0.906	0.946	**0.946**	0.945	0.824	0.862	0.835	0.846	0.662	0.715	0.686	0.689
	SERA-DIS-10	**0.942**	**0.95**	0.943	**0.955**	**0.873**	0.867	0.858	0.904	**0.721**	0.713	0.723	**0.76**
	SERA-DIS-NP-5	0.902	0.936	0.938	0.94	0.814	0.829	0.79	0.836	0.645	0.664	0.626	0.672
	SERA-DIS-NP-10	0.936	0.947	0.94	0.945	0.847	0.858	0.845	0.843	0.682	0.707	0.69	0.68
	SERA-DIS-KW-5	0.902	0.943	0.941	0.942	0.831	0.855	0.823	0.847	0.666	0.701	0.663	0.687
	SERA-DIS-KW-10	0.94	0.945	0.939	0.952	0.868	0.852	0.866	**0.907**	0.715	0.698	**0.732**	0.759
	wikiSERA-5	0.862	0.922	0.921	0.921	0.824	0.871	0.804	0.839	0.666	0.732	0.648	0.68
	wikiSERA-10	0.885	0.936	0.931	0.927	**0.876**	**0.889**	**0.862**	0.868	**0.727**	**0.756**	**0.715**	0.705
	wikiSERA-DIS-5	0.904	0.944	0.935	0.949	0.829	0.856	0.789	0.852	0.667	0.722	0.632	0.706
	wikiSERA-DIS-10	**0.94**	**0.955**	**0.945**	**0.957**	0.872	0.88	0.845	**0.886**	0.715	0.745	0.698	**0.733**

Table A.55: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.87	0.924	0.947	0.949	0.814	0.872	0.876	0.893	0.648	0.721	0.724	0.74
	SERA-10	0.907	0.936	0.942	0.945	0.881	**0.897**	0.863	0.903	0.722	**0.752**	0.71	0.758
	SERA-NP-5	0.884	0.922	0.924	0.934	0.821	0.826	0.798	0.878	0.652	0.655	0.635	0.723
	SERA-NP-10	0.9	0.932	0.944	0.942	**0.888**	0.87	**0.887**	0.872	0.728	0.719	**0.738**	0.721
	SERA-KW-5	0.872	0.921	0.944	0.946	0.82	0.869	0.865	0.89	0.654	0.716	0.7	0.732
	SERA-KW-10	0.908	0.935	0.935	0.946	0.878	0.885	0.861	0.902	0.723	0.742	0.711	0.754
	SERA-DIS-5	0.913	0.944	**0.948**	0.958	0.82	0.857	0.867	0.889	0.654	0.7	0.705	0.74
	SERA-DIS-10	**0.949**	**0.952**	0.944	**0.959**	0.884	0.877	0.875	**0.909**	**0.731**	0.718	0.731	**0.776**
	SERA-DIS-NP-5	0.918	0.945	0.94	0.951	0.807	0.858	0.769	0.848	0.634	0.688	0.6	0.689
	SERA-DIS-NP-10	0.943	0.948	0.942	0.949	0.867	0.88	0.834	0.851	0.706	0.721	0.685	0.7
	SERA-DIS-KW-5	0.908	0.941	0.943	0.954	0.826	0.855	0.86	0.887	0.666	0.694	0.693	0.74
	SERA-DIS-KW-10	0.945	0.947	0.94	0.955	0.875	0.867	0.868	**0.909**	0.721	0.712	0.714	0.768
	wikiSERA-5	0.877	0.92	0.925	0.93	0.815	0.871	0.823	0.867	0.653	0.721	0.666	0.713
	wikiSERA-10	0.905	0.938	0.938	0.938	**0.895**	**0.894**	**0.881**	0.882	**0.748**	**0.755**	0.721	0.729
	wikiSERA-DIS-5	0.916	0.941	0.94	0.954	0.819	0.844	0.807	0.871	0.653	0.695	0.639	0.723
	wikiSERA-DIS-10	**0.947**	**0.955**	**0.948**	**0.959**	0.882	0.878	0.875	**0.888**	0.731	0.735	**0.724**	**0.743**

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.87	0.923	0.947	0.949	0.817	0.873	0.879	0.894	0.653	0.723	0.729	0.742
	SERA-10	0.907	0.936	0.942	0.945	0.883	**0.899**	0.866	0.906	0.724	**0.754**	0.715	0.763
	SERA-NP-5	0.883	0.922	0.924	0.934	0.822	0.827	0.803	0.879	0.652	0.656	0.64	0.725
	SERA-NP-10	0.9	0.932	0.944	0.942	**0.89**	0.873	**0.889**	0.875	0.73	0.721	**0.74**	0.726
	SERA-KW-5	0.871	0.921	0.944	0.946	0.823	0.87	0.867	0.891	0.659	0.718	0.705	0.734
	SERA-KW-10	0.908	0.935	0.935	0.946	0.88	0.887	0.864	0.906	0.725	0.745	0.716	0.759
	SERA-DIS-5	0.912	0.944	**0.949**	0.958	0.823	0.858	0.87	0.891	0.656	0.702	0.71	0.745
	SERA-DIS-10	**0.949**	**0.952**	0.945	**0.959**	0.886	0.879	0.878	**0.913**	**0.733**	0.72	0.736	**0.781**
	SERA-DIS-NP-5	0.918	0.945	0.94	0.951	0.809	0.86	0.773	0.848	0.636	0.692	0.605	0.688
	SERA-DIS-NP-10	0.943	0.948	0.943	0.949	0.868	0.882	0.836	0.854	0.709	0.723	0.687	0.705
	SERA-DIS-KW-5	0.908	0.94	0.944	0.954	0.829	0.856	0.863	0.89	0.668	0.697	0.698	0.745
	SERA-DIS-KW-10	0.945	0.947	0.941	0.955	0.877	0.868	0.871	**0.913**	0.723	0.714	0.719	0.773
	wikiSERA-5	0.876	0.92	0.925	0.929	0.817	0.872	0.826	0.868	0.655	0.724	0.668	0.713
	wikiSERA-10	0.905	0.938	0.938	0.938	**0.897**	**0.896**	**0.883**	0.884	**0.75**	**0.757**	**0.726**	0.729
	wikiSERA-DIS-5	0.916	0.941	0.94	0.954	0.821	0.845	0.809	0.872	0.655	0.697	0.641	0.725
	wikiSERA-DIS-10	**0.947**	**0.955**	**0.948**	**0.959**	0.883	0.879	0.877	**0.889**	0.733	0.734	**0.726**	**0.742**

Table A.56: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.886	0.931	0.942	0.936	0.826	0.875	0.848	0.822	0.679	0.736	0.696	0.663
	SERA-10	0.913	0.94	0.94	0.945	0.874	**0.88**	0.835	0.874	**0.735**	**0.749**	0.675	0.721
	SERA-NP-5	0.896	0.917	0.925	0.923	0.835	0.804	0.785	0.84	0.675	0.63	0.619	0.669
	SERA-NP-10	0.907	0.934	**0.948**	0.939	0.849	0.854	**0.895**	0.854	0.7	0.698	**0.735**	0.696
	SERA-KW-5	0.888	0.928	0.938	0.934	0.83	0.865	0.827	0.833	0.679	0.72	0.671	0.665
	SERA-KW-10	0.916	0.938	0.938	0.944	0.867	0.866	0.841	**0.876**	0.727	0.727	0.682	0.72
	SERA-DIS-5	0.926	**0.95**	0.941	0.944	0.82	0.872	0.846	0.849	0.658	0.717	0.69	0.689
	SERA-DIS-10	**0.952**	0.948	0.935	**0.948**	**0.879**	0.876	0.855	0.875	**0.735**	0.721	0.715	**0.725**
	SERA-DIS-NP-5	0.928	0.94	0.937	0.938	0.848	0.832	0.775	0.823	0.678	0.667	0.623	0.659
	SERA-DIS-NP-10	0.945	0.944	0.934	0.94	0.855	0.855	0.852	0.841	0.694	0.694	0.696	0.679
	SERA-DIS-KW-5	0.921	0.947	0.937	0.94	0.827	0.861	0.826	0.863	0.666	0.706	0.678	0.704
	SERA-DIS-KW-10	0.95	0.944	0.933	0.945	**0.879**	0.855	0.854	**0.876**	0.732	0.696	0.708	0.724
	wikiSERA-5	0.885	0.925	0.925	0.924	0.821	0.851	0.786	0.817	0.669	0.702	0.626	0.648
	wikiSERA-10	0.912	0.938	**0.942**	0.936	**0.879**	**0.883**	0.853	0.841	**0.732**	**0.744**	0.696	0.675
	wikiSERA-DIS-5	0.922	0.945	0.935	0.947	0.824	0.844	0.79	0.862	0.665	0.693	0.644	**0.716**
	wikiSERA-DIS-10	**0.949**	**0.95**	**0.942**	**0.95**	0.874	0.861	**0.871**	**0.865**	0.72	0.713	**0.732**	0.71

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.886	0.931	0.943	0.936	0.828	0.875	0.85	0.823	0.682	0.737	0.701	0.666
	SERA-10	0.913	0.94	0.94	0.945	0.875	**0.882**	0.838	0.877	0.734	**0.754**	0.678	0.721
	SERA-NP-5	0.896	0.916	0.925	0.923	0.836	0.804	0.788	0.84	0.675	0.629	0.622	0.671
	SERA-NP-10	0.907	0.934	**0.948**	0.939	0.851	0.857	**0.897**	0.857	0.699	0.7	**0.737**	0.701
	SERA-KW-5	0.887	0.928	0.939	0.934	0.833	0.865	0.829	0.834	0.681	0.721	0.676	0.667
	SERA-KW-10	0.916	0.938	0.938	0.944	0.869	0.868	0.845	0.879	0.726	0.732	0.687	0.722
	SERA-DIS-5	0.926	**0.95**	0.941	0.944	0.822	0.873	0.848	0.851	0.66	0.719	0.695	0.691
	SERA-DIS-10	**0.951**	0.948	0.936	**0.948**	**0.88**	0.878	0.858	0.879	**0.737**	0.723	0.72	0.728
	SERA-DIS-NP-5	0.927	0.94	0.938	0.938	0.848	0.833	0.777	0.823	0.677	0.667	0.625	0.659
	SERA-DIS-NP-10	0.945	0.944	0.934	0.94	0.856	0.857	0.853	0.843	0.694	0.696	0.698	0.682
	SERA-DIS-KW-5	0.921	0.947	0.937	0.94	0.83	0.862	0.828	0.865	0.668	0.709	0.683	0.706
	SERA-DIS-KW-10	0.95	0.944	0.934	0.946	**0.88**	0.857	0.857	**0.88**	0.734	0.698	0.713	**0.729**
	wikiSERA-5	0.884	0.925	0.925	0.924	0.822	0.852	0.789	0.817	0.671	0.705	0.629	0.648
	wikiSERA-10	0.912	0.938	**0.942**	0.936	**0.88**	**0.885**	0.855	0.843	**0.735**	**0.744**	0.695	0.675
	wikiSERA-DIS-5	0.922	0.945	0.935	0.947	0.825	0.845	0.792	0.862	0.667	0.695	0.647	**0.715**
	wikiSERA-DIS-10	**0.949**	**0.95**	**0.942**	**0.951**	0.875	0.863	**0.872**	**0.866**	0.722	0.715	**0.732**	0.71

Table A.57: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 3 reference summaries	SERA-5	0.873	0.925	**0.95**	0.942	0.82	0.88	0.873	0.871	0.661	**0.743**	0.722	0.717
	SERA-10	0.903	0.936	0.94	0.944	0.87	**0.881**	0.861	0.892	0.725	0.737	0.707	0.741
	SERA-NP-5	0.882	0.915	0.927	0.926	0.83	0.811	0.798	0.863	0.663	0.641	0.635	0.704
	SERA-NP-10	0.898	0.931	0.943	0.936	0.865	0.866	**0.892**	0.863	0.708	0.718	0.734	0.709
	SERA-KW-5	0.874	0.923	0.946	0.939	0.83	0.869	0.859	0.863	0.67	0.721	0.709	0.701
	SERA-KW-10	0.905	0.935	0.937	0.944	0.865	0.875	0.861	0.894	0.722	0.725	0.706	0.738
	SERA-DIS-5	0.916	0.947	**0.95**	0.952	0.824	0.866	0.868	0.877	0.663	0.714	0.709	0.729
	SERA-DIS-10	**0.948**	**0.951**	0.943	**0.955**	**0.877**	0.879	0.871	0.896	**0.729**	0.728	**0.736**	0.751
	SERA-DIS-NP-5	0.917	0.942	0.943	0.945	0.829	0.842	0.796	0.842	0.659	0.673	0.636	0.684
	SERA-DIS-NP-10	0.943	0.947	0.941	0.945	0.866	0.872	0.853	0.845	0.703	0.717	0.706	0.688
	SERA-DIS-KW-5	0.911	0.944	0.945	0.949	0.821	0.86	0.863	0.868	0.658	0.704	0.706	0.713
	SERA-DIS-KW-10	0.945	0.946	0.94	0.952	0.872	0.868	0.871	**0.899**	0.72	0.715	0.727	**0.753**
	wikiSERA-5	0.873	0.924	0.929	0.926	0.827	0.871	0.818	0.854	0.671	0.728	0.66	0.701
	wikiSERA-10	0.901	0.94	0.939	0.935	**0.885**	**0.893**	0.867	0.87	**0.741**	**0.764**	0.716	0.71
	wikiSERA-DIS-5	0.914	0.944	0.941	0.952	0.824	0.856	0.812	0.867	0.665	0.713	0.656	0.717
	wikiSERA-DIS-10	**0.946**	**0.954**	**0.947**	**0.957**	0.877	0.883	**0.878**	**0.882**	0.723	0.748	**0.735**	**0.732**
		Pearson				Spearman				Kendall			
		1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
Average score with 4 reference summaries	SERA-5	0.873	0.925	**0.95**	0.942	0.823	0.881	0.875	0.872	0.666	**0.745**	0.727	0.719
	SERA-10	0.903	0.936	0.94	0.944	0.872	**0.883**	0.863	0.895	0.727	0.74	0.71	0.743
	SERA-NP-5	0.881	0.915	0.928	0.926	0.831	0.811	0.802	0.863	0.662	0.639	0.64	0.705
	SERA-NP-10	0.897	0.93	0.943	0.936	0.866	0.868	**0.893**	0.866	0.71	0.72	0.737	0.714
	SERA-KW-5	0.874	0.923	0.946	0.939	0.832	0.87	0.862	0.865	0.675	0.724	0.714	0.703
	SERA-KW-10	0.905	0.935	0.937	0.944	0.866	0.877	0.864	0.897	0.724	0.727	0.711	0.741
	SERA-DIS-5	0.916	0.947	**0.95**	0.952	0.827	0.867	0.87	0.88	0.666	0.717	0.714	0.732
	SERA-DIS-10	**0.948**	**0.951**	0.943	**0.955**	**0.879**	0.881	0.873	0.899	**0.732**	0.73	**0.741**	0.756
	SERA-DIS-NP-5	0.917	0.942	0.943	0.946	0.83	0.843	0.798	0.842	0.659	0.672	0.639	0.683
	SERA-DIS-NP-10	0.943	0.948	0.941	0.946	0.867	0.875	0.855	0.848	0.703	0.719	0.709	0.69
	SERA-DIS-KW-5	0.911	0.944	0.946	0.949	0.823	0.862	0.866	0.871	0.66	0.706	0.711	0.715
	SERA-DIS-KW-10	0.945	0.946	0.941	0.952	0.873	0.87	0.874	**0.902**	0.722	0.718	0.732	**0.758**
	wikiSERA-5	0.873	0.924	0.929	0.926	0.829	0.872	0.821	0.854	0.673	0.73	0.662	0.701
	wikiSERA-10	0.901	0.939	0.939	0.935	**0.887**	**0.894**	0.869	0.872	**0.743**	**0.764**	0.719	0.71
	wikiSERA-DIS-5	0.914	0.944	0.941	0.952	0.825	0.857	0.814	0.868	0.667	0.715	0.658	0.717
	wikiSERA-DIS-10	**0.946**	**0.954**	**0.947**	**0.957**	0.878	0.884	**0.88**	**0.883**	0.726	0.748	**0.734**	**0.732**

Table A.58: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Pyramid on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.2.8 Correlation of SERA and wikiSERA with Responsiveness on TAC2009/Wikipedia

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.77	0.822	0.845	**0.862**	0.669	**0.742**	0.742	**0.799**	0.502	0.568	0.573	**0.62**
SERA-10	0.824	**0.842**	**0.846**	0.85	0.765	0.732	0.735	0.791	0.601	0.562	0.568	0.615
SERA-NP-5	0.799	0.796	0.81	0.839	0.708	0.646	0.641	0.751	0.552	0.495	0.488	0.587
SERA-NP-10	0.822	0.841	0.838	0.829	**0.778**	0.736	0.715	0.725	0.603	**0.574**	0.546	0.555
SERA-KW-5	0.766	0.815	0.834	0.854	0.676	**0.742**	0.717	0.769	0.507	0.568	0.551	0.598
SERA-KW-10	0.819	0.84	0.835	0.846	0.77	0.729	0.73	0.797	**0.605**	0.553	0.561	0.618
SERA-DIS-5	0.779	0.804	0.823	0.841	0.653	0.708	**0.748**	0.796	0.488	0.532	**0.579**	0.617
SERA-DIS-10	0.816	0.811	0.812	0.819	0.715	0.7	0.724	0.779	0.549	0.527	0.546	0.595
SERA-DIS-NP-5	0.81	0.799	0.8	0.84	0.708	0.664	0.632	0.756	0.54	0.499	0.475	0.588
SERA-DIS-NP-10	**0.833**	0.811	0.802	0.812	0.755	0.725	0.687	0.721	0.575	0.546	0.528	0.559
SERA-DIS-KW-5	0.772	0.797	0.812	0.833	0.643	0.714	0.728	0.773	0.477	0.539	0.557	0.6
SERA-DIS-KW-10	0.806	0.802	0.802	0.809	0.704	0.696	0.718	0.774	0.538	0.522	0.534	0.599
wikiSERA-5	0.796	0.823	0.831	0.85	0.712	0.709	0.74	0.764	0.53	0.555	0.563	0.597
wikiSERA-10	0.831	**0.869**	**0.845**	**0.862**	**0.767**	0.762	**0.755**	0.783	**0.601**	0.602	**0.59**	**0.634**
wikiSERA-DIS-5	0.813	0.806	0.807	0.842	0.692	0.686	0.716	0.785	0.517	0.532	0.548	0.606
wikiSERA-DIS-10	**0.842**	0.831	0.805	0.843	0.751	**0.777**	0.717	**0.788**	0.584	**0.608**	0.549	0.622

Table A.59: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.807	0.829	0.808	0.832	0.709	0.722	0.634	0.689	0.52	0.544	0.485	0.54
SERA-10	0.851	**0.843**	0.821	**0.834**	0.78	0.741	0.676	0.75	**0.619**	0.566	0.507	0.589
SERA-NP-5	0.808	0.791	0.789	0.82	0.734	0.654	0.655	0.722	0.555	0.494	0.503	0.562
SERA-NP-10	0.829	0.836	**0.835**	0.83	0.785	0.735	**0.761**	0.725	0.597	0.575	**0.595**	0.552
SERA-KW-5	0.801	0.824	0.802	0.83	0.714	0.712	0.64	0.684	0.534	0.537	0.488	0.538
SERA-KW-10	**0.857**	0.836	0.809	0.828	0.794	0.741	0.678	0.745	0.618	0.57	0.509	0.573
SERA-DIS-5	0.812	0.803	0.77	0.808	0.719	0.713	0.609	0.717	0.521	0.532	0.46	0.549
SERA-DIS-10	0.831	0.792	0.758	0.793	0.792	**0.744**	0.664	**0.786**	0.617	**0.585**	0.51	**0.611**
SERA-DIS-NP-5	0.807	0.766	0.754	0.781	0.725	0.679	0.607	0.662	0.55	0.515	0.469	0.498
SERA-DIS-NP-10	0.806	0.771	0.747	0.768	0.773	0.743	0.694	0.698	0.581	0.581	0.532	0.537
SERA-DIS-KW-5	0.802	0.799	0.765	0.805	0.719	0.703	0.602	0.709	0.522	0.522	0.451	0.542
SERA-DIS-KW-10	0.83	0.784	0.751	0.788	**0.796**	0.727	0.648	0.757	0.61	0.572	0.498	0.577
wikiSERA-5	0.818	0.818	0.777	**0.823**	0.733	0.697	0.568	0.645	0.553	0.537	0.426	0.501
wikiSERA-10	**0.853**	**0.855**	**0.831**	0.822	**0.794**	0.743	**0.684**	0.672	**0.612**	0.579	**0.53**	0.51
wikiSERA-DIS-5	0.816	0.808	0.757	0.814	0.718	0.701	0.567	0.709	0.526	0.537	0.424	0.548
wikiSERA-DIS-10	0.831	0.813	0.769	0.797	0.772	**0.746**	0.636	**0.718**	0.579	**0.58**	0.488	**0.553**

Table A.60: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_2$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.804	0.806	0.808	0.751	0.699	**0.703**	0.606	0.542	0.53	**0.551**	0.473	0.41
SERA-10	0.814	0.809	0.798	**0.825**	0.701	0.655	0.676	**0.745**	0.539	0.507	0.519	**0.575**
SERA-NP-5	0.807	0.755	**0.82**	0.767	0.721	0.585	0.683	0.645	0.536	0.441	0.521	0.49
SERA-NP-10	0.794	**0.814**	0.797	0.807	0.659	0.695	**0.698**	0.726	0.498	0.519	0.522	0.561
SERA-KW-5	0.797	0.793	0.79	0.735	0.693	0.688	0.579	0.527	0.519	0.537	0.447	0.399
SERA-KW-10	0.813	0.799	0.798	0.818	0.703	0.647	0.693	0.736	0.54	0.502	**0.538**	0.572
SERA-DIS-5	**0.816**	0.792	0.769	0.722	0.713	0.7	0.59	0.585	0.542	0.545	0.449	0.445
SERA-DIS-10	0.811	0.767	0.756	0.755	0.697	0.67	0.681	0.704	0.533	0.505	0.527	0.541
SERA-DIS-NP-5	**0.816**	0.754	0.784	0.765	**0.734**	0.572	0.647	0.643	**0.556**	0.424	0.494	0.476
SERA-DIS-NP-10	0.803	0.767	0.746	0.765	0.686	0.636	0.659	0.674	0.507	0.473	0.498	0.513
SERA-DIS-KW-5	0.803	0.785	0.765	0.712	0.705	0.69	0.576	0.583	0.532	0.536	0.441	0.437
SERA-DIS-KW-10	0.806	0.76	0.756	0.754	0.7	0.666	0.694	0.713	0.529	0.505	0.534	0.552
wikiSERA-5	0.803	0.81	0.785	0.781	**0.715**	0.677	0.607	0.611	0.543	0.522	0.466	0.468
wikiSERA-10	**0.816**	**0.823**	**0.801**	**0.795**	0.707	0.685	0.657	0.667	**0.544**	0.533	0.514	0.507
wikiSERA-DIS-5	0.81	0.797	0.76	0.763	0.708	0.684	0.613	0.644	0.533	0.52	0.465	0.483
wikiSERA-DIS-10	0.812	0.782	0.755	0.762	0.706	**0.686**	**0.681**	**0.681**	0.53	**0.54**	**0.524**	**0.515**

Table A.61: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_3$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.82	0.811	0.816	0.835	0.688	0.727	0.722	**0.788**	0.534	0.562	0.542	0.61
SERA-10	**0.83**	**0.85**	0.811	**0.839**	0.747	0.756	0.696	0.777	0.577	0.583	0.535	**0.613**
SERA-NP-5	0.825	0.836	0.827	0.824	0.741	0.749	0.656	0.726	0.575	0.573	0.505	0.556
SERA-NP-10	**0.83**	0.846	**0.84**	0.828	0.765	0.766	**0.777**	0.743	0.591	**0.601**	**0.615**	0.585
SERA-KW-5	0.82	0.804	0.811	0.819	0.694	0.729	0.699	0.779	0.541	0.57	0.526	0.604
SERA-KW-10	0.825	0.847	0.806	0.831	0.735	**0.769**	0.708	0.76	0.563	0.595	0.546	0.598
SERA-DIS-5	0.817	0.783	0.769	0.792	0.699	0.693	0.728	0.759	0.538	0.513	0.542	0.58
SERA-DIS-10	0.804	0.78	0.747	0.774	0.757	0.747	0.739	0.769	**0.595**	0.576	0.565	0.6
SERA-DIS-NP-5	0.818	0.819	0.795	0.8	0.766	0.756	0.661	0.719	0.585	0.573	0.502	0.545
SERA-DIS-NP-10	0.797	0.79	0.767	0.769	**0.77**	0.755	0.732	0.734	**0.595**	0.584	0.58	0.579
SERA-DIS-KW-5	0.806	0.773	0.761	0.778	0.694	0.687	0.719	0.743	0.534	0.517	0.547	0.559
SERA-DIS-KW-10	0.789	0.771	0.746	0.765	0.755	0.745	0.747	0.748	0.592	0.573	0.57	0.58
wikiSERA-5	0.821	0.821	0.808	**0.835**	0.69	0.71	0.688	**0.779**	0.523	0.538	0.536	**0.615**
wikiSERA-10	**0.842**	**0.843**	**0.823**	**0.835**	**0.788**	**0.754**	0.747	0.728	**0.619**	**0.581**	0.567	0.557
wikiSERA-DIS-5	0.814	0.792	0.773	0.804	0.68	0.7	0.71	0.772	0.509	0.531	0.527	0.599
wikiSERA-DIS-10	0.801	0.785	0.761	0.775	0.766	0.738	**0.768**	0.714	0.592	0.573	**0.595**	0.549

Table A.62: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_4$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.802	0.834	**0.844**	0.836	0.697	0.747	0.71	0.726	0.518	0.573	0.551	0.56
SERA-10	0.837	**0.843**	0.833	**0.843**	0.759	0.744	0.721	0.78	0.594	0.573	0.553	**0.6**
SERA-NP-5	0.815	0.797	0.826	0.828	0.751	0.642	0.71	0.745	0.574	0.475	0.551	0.574
SERA-NP-10	0.825	0.842	0.837	0.832	0.76	**0.751**	**0.757**	0.738	0.58	0.575	**0.58**	0.567
SERA-KW-5	0.797	0.827	0.835	0.83	0.698	0.736	0.687	0.72	0.517	0.561	0.526	0.553
SERA-KW-10	**0.838**	0.838	0.826	0.838	**0.766**	0.731	0.726	0.774	**0.599**	0.563	0.55	**0.6**
SERA-DIS-5	0.812	0.812	0.806	0.808	0.707	0.726	0.703	0.749	0.528	0.554	0.549	0.573
SERA-DIS-10	0.827	0.798	0.783	0.795	0.75	0.748	0.73	**0.782**	0.577	**0.579**	0.557	**0.6**
SERA-DIS-NP-5	0.82	0.789	0.796	0.81	0.733	0.669	0.672	0.733	0.559	0.502	0.514	0.563
SERA-DIS-NP-10	0.821	0.792	0.774	0.787	0.753	0.728	0.707	0.728	0.568	0.548	0.54	0.556
SERA-DIS-KW-5	0.801	0.808	0.8	0.802	0.71	0.729	0.692	0.751	0.53	0.548	0.534	0.571
SERA-DIS-KW-10	0.822	0.79	0.777	0.789	0.746	0.727	0.74	0.781	0.569	0.553	0.569	0.599
wikiSERA-5	0.814	0.831	0.815	0.833	0.72	0.729	0.685	0.711	0.539	0.565	0.529	0.552
wikiSERA-10	**0.841**	**0.861**	**0.839**	**0.836**	**0.778**	**0.766**	**0.732**	0.732	**0.6**	**0.599**	**0.574**	0.571
wikiSERA-DIS-5	0.822	0.816	0.788	0.82	0.729	0.732	0.659	**0.75**	0.55	0.561	0.507	**0.576**
wikiSERA-DIS-10	0.835	0.816	0.783	0.806	0.765	0.762	0.698	0.749	0.588	0.596	0.536	0.573

Table A.63: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.808	0.833	0.844	**0.86**	0.69	0.753	0.752	0.811	0.513	0.573	0.585	0.634
SERA-10	**0.845**	**0.856**	0.839	0.849	0.783	**0.788**	0.732	0.786	0.62	0.612	0.56	0.61
SERA-NP-5	0.825	0.823	0.824	0.844	0.754	0.703	0.692	0.767	0.58	0.527	0.53	0.589
SERA-NP-10	0.838	0.851	**0.85**	0.839	**0.8**	0.78	**0.772**	0.746	**0.622**	0.608	**0.605**	0.581
SERA-KW-5	0.805	0.829	0.838	0.853	0.706	0.753	0.735	0.808	0.531	0.571	0.57	0.628
SERA-KW-10	0.844	0.852	0.829	0.844	0.781	0.784	0.731	0.786	0.612	**0.619**	0.563	0.608
SERA-DIS-5	0.812	0.809	0.802	0.827	0.703	0.732	0.755	**0.814**	0.526	0.554	0.581	**0.637**
SERA-DIS-10	0.826	0.802	0.779	0.801	0.767	0.76	0.744	0.791	0.595	0.576	0.569	0.616
SERA-DIS-NP-5	0.824	0.81	0.796	0.82	0.739	0.745	0.665	0.74	0.564	0.565	0.509	0.559
SERA-DIS-NP-10	0.821	0.797	0.78	0.789	0.783	0.769	0.721	0.738	0.598	0.592	0.561	0.575
SERA-DIS-KW-5	0.803	0.804	0.796	0.819	0.701	0.735	0.745	0.802	0.527	0.554	0.575	0.623
SERA-DIS-KW-10	0.818	0.794	0.773	0.793	0.758	0.743	0.739	0.797	0.585	0.567	0.561	0.619
wikiSERA-5	0.821	0.833	0.822	**0.851**	0.715	0.764	0.725	0.76	0.536	0.602	0.557	0.595
wikiSERA-10	**0.852**	**0.866**	**0.846**	0.848	**0.801**	**0.8**	**0.761**	0.753	**0.629**	**0.633**	**0.584**	0.584
wikiSERA-DIS-5	0.824	0.814	0.791	0.831	0.715	0.733	0.705	**0.784**	0.536	0.574	0.541	**0.611**
wikiSERA-DIS-10	0.834	0.816	0.784	0.81	0.774	0.779	0.741	0.759	0.6	0.608	0.564	0.59

Table A.64: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_4$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.818	0.827	0.83	0.826	0.713	0.73	0.71	0.715	0.538	0.561	0.553	0.556
SERA-10	**0.84**	**0.844**	0.819	**0.838**	0.765	**0.757**	0.701	**0.771**	0.595	**0.599**	0.541	**0.606**
SERA-NP-5	0.823	0.807	0.827	0.818	0.744	0.679	0.701	0.731	0.559	0.503	0.543	0.565
SERA-NP-10	0.826	0.84	**0.833**	0.831	0.746	0.752	**0.784**	0.757	0.572	0.579	**0.617**	0.594
SERA-KW-5	0.814	0.819	0.822	0.816	0.717	0.724	0.678	0.723	0.539	0.558	0.523	0.557
SERA-KW-10	**0.84**	0.837	0.813	0.832	0.766	0.744	0.707	0.764	**0.598**	0.581	0.545	0.594
SERA-DIS-5	0.821	0.801	0.783	0.791	0.714	0.736	0.706	0.741	0.538	0.565	0.542	0.571
SERA-DIS-10	0.821	0.785	0.759	0.779	**0.767**	0.751	0.726	0.766	0.596	0.573	0.557	0.594
SERA-DIS-NP-5	0.822	0.794	0.79	0.794	0.758	0.703	0.676	0.721	0.576	0.525	0.534	0.55
SERA-DIS-NP-10	0.808	0.783	0.759	0.774	0.754	0.736	0.734	0.74	0.569	0.565	0.575	0.576
SERA-DIS-KW-5	0.809	0.795	0.779	0.783	0.712	0.725	0.687	0.754	0.538	0.559	0.527	0.577
SERA-DIS-KW-10	0.814	0.778	0.756	0.774	0.759	0.735	0.73	0.765	0.585	0.559	0.563	0.595
wikiSERA-5	0.822	0.831	0.806	**0.826**	0.716	0.719	0.651	0.696	0.541	0.557	0.513	0.533
wikiSERA-10	**0.844**	**0.851**	**0.828**	0.825	**0.783**	**0.756**	0.721	0.704	**0.605**	**0.595**	**0.561**	0.544
wikiSERA-DIS-5	0.82	0.811	0.775	0.805	0.724	0.731	0.648	**0.742**	0.548	0.567	0.505	**0.567**
wikiSERA-DIS-10	0.82	0.8	0.767	0.784	0.764	**0.756**	**0.723**	0.723	0.587	0.59	0.556	0.557

Table A.65: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

	Pearson				Spearman				Kendall			
	1,778,742	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**	**1,778,742**	**30,000**	**15,000**	**10,000**
SERA-5	0.81	0.832	**0.843**	0.843	0.692	0.746	0.736	0.768	0.517	0.576	0.575	0.592
SERA-10	**0.839**	**0.849**	0.832	**0.845**	0.759	0.759	0.733	**0.784**	0.593	**0.596**	0.562	**0.607**
SERA-NP-5	0.824	0.811	0.831	0.831	0.748	0.67	0.702	0.749	0.567	0.5	0.548	0.573
SERA-NP-10	0.831	0.845	0.841	0.835	**0.771**	**0.76**	**0.781**	0.759	0.593	0.59	**0.609**	0.592
SERA-KW-5	0.806	0.825	0.836	0.834	0.705	0.746	0.709	0.761	0.53	0.572	0.553	0.588
SERA-KW-10	**0.839**	0.843	0.825	0.839	0.761	0.75	0.735	0.771	**0.598**	0.588	0.567	0.588
SERA-DIS-5	0.816	0.808	0.801	0.809	0.703	0.733	0.736	0.778	0.527	0.556	0.569	0.602
SERA-DIS-10	0.824	0.795	0.775	0.791	0.762	0.753	0.743	0.781	0.585	0.577	0.569	0.603
SERA-DIS-NP-5	0.825	0.801	0.799	0.811	0.745	0.7	0.686	0.733	0.569	0.526	0.532	0.563
SERA-DIS-NP-10	0.818	0.793	0.774	0.785	0.77	0.751	0.73	0.746	0.582	0.569	0.564	0.579
SERA-DIS-KW-5	0.805	0.802	0.795	0.801	0.696	0.73	0.729	0.773	0.519	0.55	0.561	0.596
SERA-DIS-KW-10	0.817	0.788	0.771	0.785	0.752	0.738	0.738	0.782	0.576	0.558	0.565	**0.607**
wikiSERA-5	0.82	0.833	0.819	0.838	0.723	0.74	0.693	0.737	0.553	0.575	0.542	0.57
wikiSERA-10	**0.845**	**0.86**	**0.839**	**0.839**	**0.787**	**0.777**	0.738	0.737	**0.611**	**0.615**	**0.574**	0.571
wikiSERA-DIS-5	0.823	0.814	0.788	0.819	0.719	0.736	0.697	**0.768**	0.542	0.568	0.546	**0.592**
wikiSERA-DIS-10	0.83	0.81	0.779	0.8	0.773	0.77	**0.744**	0.748	0.601	0.6	0.568	0.577

Table A.66: Correlation coefficients, in terms of Pearson, Spearman and Kendall of SERA and wikiSERA, with Responsiveness on TAC2009/Wikipedia dataset using the reference summary $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, $\mathcal{A}_4$

A.3 SummTriver

A.3.1 Correlation of SummTriver with Pyramid on TAC 2008

		Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
		Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
600 summaries per corpus	3 models	JS $\mathcal{T}_m$	-0.705	-0.735	-0.883	-0.885	-0.871	-0.874	-0.846	-0.854	-0.823	-0.831
		sJS $\mathcal{T}_m$	-0.697	-0.731	-0.879	-0.883	-0.866	-0.871	-0.844	-0.853	-0.823	-0.832
		KL $\mathcal{T}_m$	-0.347	-0.450	-0.543	-0.647	-0.615	-0.684	-0.578	-0.668	-0.639	-0.701
		JS $\mathcal{T}_c$	-0.393	-0.412	-0.804	-0.807	-0.813	-0.816	-0.827	-0.831	-0.788	-0.797
		sJS $\mathcal{T}_c$	-0.393	-0.412	-0.804	-0.807	-0.812	-0.815	-0.826	-0.830	-0.788	-0.797
		KL $\mathcal{T}_c$	0.065	0.047	0.107	0.127	0.045	0.212	0.011	-0.049	0.047	0.109
	4 models	JS $\mathcal{T}_m$	-0.703	-0.733	-0.881	-0.883	-0.872	-0.875	-0.848	-0.855	-0.824	-0.831
		sJS $\mathcal{T}_m$	-0.695	-0.729	-0.877	-0.881	-0.867	-0.871	-0.845	-0.854	-0.824	-0.832
		KL $\mathcal{T}_m$	-0.344	-0.447	-0.539	-0.643	-0.616	-0.685	-0.583	-0.673	-0.643	-0.703
		JS $\mathcal{T}_c$	-0.392	-0.410	-0.802	-0.805	-0.814	-0.817	-0.828	-0.832	-0.788	-0.797
		sJS $\mathcal{T}_c$	-0.392	-0.410	-0.801	-0.805	-0.813	-0.816	-0.828	-0.832	-0.788	-0.797
		KL $\mathcal{T}_c$	0.068	0.049	0.114	0.132	0.048	0.213	0.013	-0.048	0.045	0.107
900 summaries per corpus	3 models	JS $\mathcal{T}_m$	-0.807	-0.830	-0.799	-0.808	-0.869	-0.878	-0.824	-0.836	-0.880	-0.889
		sJS $\mathcal{T}_m$	-0.807	-0.832	-0.794	-0.804	-0.863	-0.875	-0.820	-0.833	-0.876	-0.885
		KL $\mathcal{T}_m$	-0.623	-0.704	-0.552	-0.605	-0.620	-0.699	-0.614	-0.663	-0.609	-0.694
		JS $\mathcal{T}_c$	-0.534	-0.553	-0.710	-0.714	-0.825	-0.832	-0.792	-0.799	-0.854	-0.858
		sJS $\mathcal{T}_c$	-0.534	-0.553	-0.709	-0.714	-0.824	-0.831	-0.791	-0.799	-0.853	-0.857
		KL $\mathcal{T}_c$	-0.269	-0.241	-0.006	0.153	0.177	0.183	0.103	0.254	-0.274	-0.216
	4 models	JS $\mathcal{T}_m$	-0.809	-0.832	-0.801	-0.809	-0.868	-0.878	-0.822	-0.834	-0.882	-0.890
		sJS $\mathcal{T}_m$	-0.809	-0.834	-0.796	-0.805	-0.863	-0.874	-0.818	-0.831	-0.877	-0.887
		KL $\mathcal{T}_m$	-0.625	-0.706	-0.554	-0.608	-0.622	-0.700	-0.613	-0.663	-0.611	-0.696
		JS $\mathcal{T}_c$	-0.537	-0.556	-0.712	-0.716	-0.825	-0.832	-0.789	-0.797	-0.855	-0.859
		sJS $\mathcal{T}_c$	-0.537	-0.556	-0.712	-0.716	-0.824	-0.832	-0.788	-0.796	-0.854	-0.858
		KL $\mathcal{T}_c$	-0.269	-0.243	-0.005	0.157	0.188	0.192	0.106	0.257	-0.273	-0.216

Table A.67: Correlation coefficients in terms of Pearson of SummTriver with Pyramid 3 reference summaries and Pyramid 4 reference summaries on TAC 2008 dataset, using 600 and 900 summaries per corpus

		Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
		Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
600 summaries per corpus	3 models	JS $\mathcal{T}_m$	-0.676	-0.709	-0.875	-0.876	-0.834	-0.838	-0.826	-0.833	-0.807	-0.815
		sJS $\mathcal{T}_m$	-0.681	-0.718	-0.868	-0.865	-0.832	-0.836	-0.820	-0.827	-0.811	-0.817
		KL $\mathcal{T}_m$	-0.393	-0.475	-0.581	-0.651	-0.635	-0.689	-0.584	-0.675	-0.630	-0.696
		JS $\mathcal{T}_c$	-0.420	-0.437	-0.788	-0.793	-0.783	-0.787	-0.809	-0.806	-0.780	-0.796
		sJS $\mathcal{T}_c$	-0.420	-0.437	-0.788	-0.793	-0.783	-0.787	-0.810	-0.806	-0.780	-0.796
		KL $\mathcal{T}_c$	0.076	0.006	0.052	0.073	0.016	0.183	-0.040	-0.142	0.093	0.142
	4 models	JS $\mathcal{T}_m$	-0.679	-0.711	-0.870	-0.871	-0.836	-0.840	-0.827	-0.834	-0.800	-0.808
		sJS $\mathcal{T}_m$	-0.685	-0.721	-0.863	-0.861	-0.833	-0.837	-0.822	-0.828	-0.806	-0.811
		KL $\mathcal{T}_m$	-0.395	-0.476	-0.572	-0.643	-0.633	-0.688	-0.587	-0.677	-0.627	-0.690
		JS $\mathcal{T}_c$	-0.424	-0.441	-0.782	-0.788	-0.784	-0.788	-0.810	-0.807	-0.774	-0.790
		sJS $\mathcal{T}_c$	-0.424	-0.441	-0.782	-0.788	-0.784	-0.788	-0.811	-0.807	-0.774	-0.790
		KL $\mathcal{T}_c$	0.080	0.004	0.060	0.080	0.020	0.180	-0.039	-0.141	0.097	0.143
900 summaries per corpus	3 models	JS $\mathcal{T}_m$	-0.756	-0.791	-0.745	-0.744	-0.831	-0.841	-0.777	-0.795	-0.813	-0.827
		sJS $\mathcal{T}_m$	-0.755	-0.792	-0.740	-0.739	-0.830	-0.838	-0.775	-0.793	-0.813	-0.822
		KL $\mathcal{T}_m$	-0.627	-0.703	-0.546	-0.583	-0.642	-0.735	-0.609	-0.669	-0.636	-0.700
		JS $\mathcal{T}_c$	-0.565	-0.574	-0.681	-0.677	-0.810	-0.812	-0.744	-0.745	-0.799	-0.805
		sJS $\mathcal{T}_c$	-0.565	-0.574	-0.679	-0.677	-0.810	-0.812	-0.746	-0.745	-0.799	-0.805
		KL $\mathcal{T}_c$	-0.192	-0.196	-0.018	0.101	0.221	0.241	0.066	0.152	-0.275	-0.168
	4 models	JS $\mathcal{T}_m$	-0.755	-0.791	-0.748	-0.746	-0.839	-0.849	-0.772	-0.791	-0.813	-0.826
		sJS $\mathcal{T}_m$	-0.755	-0.792	-0.743	-0.742	-0.837	-0.846	-0.771	-0.789	-0.814	-0.821
		KL $\mathcal{T}_m$	-0.632	-0.707	-0.550	-0.588	-0.649	-0.742	-0.606	-0.665	-0.640	-0.701
		JS $\mathcal{T}_c$	-0.567	-0.575	-0.685	-0.681	-0.819	-0.821	-0.737	-0.737	-0.799	-0.805
		sJS $\mathcal{T}_c$	-0.567	-0.575	-0.683	-0.681	-0.819	-0.821	-0.739	-0.737	-0.799	-0.805
		KL $\mathcal{T}_c$	-0.194	-0.197	-0.015	0.109	0.220	0.239	0.070	0.152	-0.269	-0.164

Table A.68: Correlation coefficients in terms of Spearman of SummTriver with Pyramid 3 reference summaries and Pyramid 4 reference summaries on TAC 2008 dataset, using 600 and 900 summaries per corpus

		Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
		Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
600 summaries per corpus	3 models	JS $\mathcal{T}_m$	-0.495	-0.519	-0.699	-0.699	-0.645	-0.648	-0.636	-0.647	-0.606	-0.614
		sJS $\mathcal{T}_m$	-0.493	-0.529	-0.693	-0.693	-0.639	-0.645	-0.629	-0.643	-0.613	-0.616
		KL $\mathcal{T}_m$	-0.270	-0.330	-0.435	-0.492	-0.464	-0.514	-0.405	-0.474	-0.443	-0.497
		JS $\mathcal{T}_c$	-0.295	-0.315	-0.600	-0.610	-0.579	-0.583	-0.614	-0.613	-0.561	-0.586
		sJS $\mathcal{T}_c$	-0.295	-0.315	-0.600	-0.610	-0.579	-0.583	-0.615	-0.613	-0.562	-0.587
		KL $\mathcal{T}_c$	0.057	0.004	0.043	0.050	-0.005	0.111	-0.024	-0.107	0.053	0.091
	4 models	JS $\mathcal{T}_m$	-0.493	-0.522	-0.690	-0.690	-0.643	-0.645	-0.638	-0.649	-0.601	-0.609
		sJS $\mathcal{T}_m$	-0.491	-0.532	-0.687	-0.684	-0.637	-0.643	-0.631	-0.645	-0.608	-0.612
		KL $\mathcal{T}_m$	-0.273	-0.333	-0.431	-0.488	-0.464	-0.511	-0.407	-0.476	-0.441	-0.492
		JS $\mathcal{T}_c$	-0.295	-0.314	-0.593	-0.604	-0.581	-0.585	-0.616	-0.613	-0.561	-0.584
		sJS $\mathcal{T}_c$	-0.295	-0.314	-0.593	-0.604	-0.581	-0.585	-0.618	-0.613	-0.562	-0.585
		KL $\mathcal{T}_c$	0.066	0.005	0.044	0.056	-0.001	0.113	-0.024	-0.105	0.053	0.089
900 summaries per corpus	3 models	JS $\mathcal{T}_m$	-0.572	-0.606	-0.567	-0.561	-0.639	-0.655	-0.593	-0.609	-0.630	-0.643
		sJS $\mathcal{T}_m$	-0.569	-0.607	-0.561	-0.552	-0.638	-0.653	-0.587	-0.604	-0.627	-0.637
		KL $\mathcal{T}_m$	-0.442	-0.510	-0.378	-0.402	-0.451	-0.531	-0.425	-0.479	-0.460	-0.510
		JS $\mathcal{T}_c$	-0.400	-0.406	-0.499	-0.489	-0.612	-0.612	-0.561	-0.563	-0.606	-0.613
		sJS $\mathcal{T}_c$	-0.400	-0.406	-0.497	-0.489	-0.610	-0.612	-0.563	-0.563	-0.604	-0.612
		KL $\mathcal{T}_c$	-0.132	-0.126	-0.014	0.072	0.153	0.166	0.027	0.103	-0.183	-0.123
	4 models	JS $\mathcal{T}_m$	-0.579	-0.615	-0.568	-0.560	-0.644	-0.660	-0.591	-0.607	-0.629	-0.637
		sJS $\mathcal{T}_m$	-0.579	-0.614	-0.560	-0.554	-0.643	-0.658	-0.585	-0.602	-0.626	-0.633
		KL $\mathcal{T}_m$	-0.449	-0.515	-0.382	-0.408	-0.456	-0.535	-0.425	-0.476	-0.459	-0.511
		JS $\mathcal{T}_c$	-0.400	-0.406	-0.503	-0.493	-0.616	-0.616	-0.551	-0.554	-0.604	-0.612
		sJS $\mathcal{T}_c$	-0.400	-0.406	-0.500	-0.493	-0.615	-0.616	-0.554	-0.554	-0.603	-0.610
		KL $\mathcal{T}_c$	-0.128	-0.126	-0.013	0.078	0.151	0.162	0.027	0.101	-0.180	-0.119

Table A.69: Correlation coefficients in terms of Kendall of SummTriver with Pyramid 3 reference summaries and Pyramid 4 reference summaries on TAC 2008 dataset, using 600 and 900 summaries per corpus

A.3.2 Correlation of SummTriver with Responsiveness on TAC 2008

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
600 summaries	JS $\mathcal{T}_m$	-0.645	-0.671	-0.781	-0.775	-0.821	-0.828	-0.744	-0.749	-0.815	-0.820
	sJS $\mathcal{T}_m$	-0.639	-0.668	-0.779	-0.774	-0.816	-0.824	-0.744	-0.749	-0.817	-0.822
	KL $\mathcal{T}_m$	-0.314	-0.408	-0.509	-0.593	-0.568	-0.644	-0.540	-0.603	-0.652	-0.706
	JS $\mathcal{T}_c$	-0.391	-0.409	-0.730	-0.725	-0.754	-0.761	-0.727	-0.730	-0.768	-0.771
	sJS $\mathcal{T}_c$	-0.391	-0.409	-0.730	-0.725	-0.753	-0.760	-0.727	-0.730	-0.768	-0.771
	KL $\mathcal{T}_c$	0.100	0.095	0.154	0.189	0.141	0.233	-0.103	-0.107	-0.089	-0.025
900 summaries	JS $\mathcal{T}_m$	-0.645	-0.671	-0.781	-0.775	-0.821	-0.828	-0.744	-0.749	-0.815	-0.820
	sJS $\mathcal{T}_m$	-0.639	-0.668	-0.779	-0.774	-0.816	-0.824	-0.744	-0.749	-0.817	-0.822
	KL $\mathcal{T}_m$	-0.314	-0.408	-0.509	-0.593	-0.568	-0.644	-0.540	-0.603	-0.652	-0.706
	JS $\mathcal{T}_c$	-0.391	-0.409	-0.730	-0.725	-0.754	-0.761	-0.727	-0.730	-0.768	-0.771
	sJS $\mathcal{T}_c$	-0.391	-0.409	-0.730	-0.725	-0.753	-0.760	-0.727	-0.730	-0.768	-0.771
	KL $\mathcal{T}_c$	0.100	0.095	0.154	0.189	0.141	0.233	-0.103	-0.107	-0.089	-0.025

Table A.70: Correlation coefficients in terms of Pearson of SummTriver with Responsiveness on TAC 2008 dataset, using 600 and 900 summary per corpus

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
600 summaries	JS $\mathcal{T}_m$	-0.629	-0.659	-0.780	-0.771	-0.801	-0.802	-0.696	-0.712	-0.776	-0.800
	sJS $\mathcal{T}_m$	-0.632	-0.667	-0.773	-0.759	-0.802	-0.798	-0.704	-0.708	-0.783	-0.801
	KL $\mathcal{T}_m$	-0.337	-0.417	-0.555	-0.612	-0.615	-0.672	-0.533	-0.593	-0.665	-0.736
	JS $\mathcal{T}_c$	-0.401	-0.425	-0.714	-0.709	-0.751	-0.762	-0.699	-0.698	-0.752	-0.760
	sJS $\mathcal{T}_c$	-0.401	-0.425	-0.714	-0.709	-0.751	-0.762	-0.699	-0.698	-0.757	-0.761
	KL $\mathcal{T}_c$	0.112	0.060	0.102	0.080	0.077	0.183	-0.083	-0.126	0.015	0.053
900 summaries	JS $\mathcal{T}_m$	-0.698	-0.714	-0.718	-0.724	-0.779	-0.784	-0.590	-0.602	-0.786	-0.801
	sJS $\mathcal{T}_m$	-0.701	-0.722	-0.717	-0.725	-0.778	-0.782	-0.587	-0.603	-0.789	-0.797
	KL $\mathcal{T}_m$	-0.606	-0.668	-0.594	-0.633	-0.649	-0.712	-0.502	-0.547	-0.620	-0.695
	JS $\mathcal{T}_c$	-0.548	-0.552	-0.690	-0.689	-0.801	-0.800	-0.574	-0.576	-0.768	-0.777
	sJS $\mathcal{T}_c$	-0.548	-0.552	-0.689	-0.689	-0.801	-0.800	-0.576	-0.576	-0.769	-0.777
	KL $\mathcal{T}_c$	-0.136	-0.120	-0.023	0.184	0.181	0.178	0.152	0.172	-0.177	-0.134

Table A.71: Correlation coefficients in terms of Spearman of SummTriver with Responsiveness on TAC 2008 dataset, using 600 and 900 summary per corpus

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
600 summaries	JS $\mathcal{T}_m$	-0.451	-0.483	-0.589	-0.576	-0.604	-0.602	-0.512	-0.523	-0.572	-0.590
	sJS $\mathcal{T}_m$	-0.454	-0.488	-0.581	-0.565	-0.605	-0.595	-0.514	-0.517	-0.577	-0.590
	KL $\mathcal{T}_m$	-0.233	-0.296	-0.409	-0.443	-0.437	-0.483	-0.382	-0.412	-0.483	-0.529
	JS $\mathcal{T}_c$	-0.280	-0.301	-0.529	-0.526	-0.562	-0.568	-0.515	-0.519	-0.533	-0.539
	sJS $\mathcal{T}_c$	-0.280	-0.301	-0.529	-0.526	-0.562	-0.568	-0.514	-0.519	-0.539	-0.540
	KL $\mathcal{T}_c$	0.071	0.047	0.068	0.072	0.035	0.113	-0.056	-0.081	0.022	0.026
900 summaries	JS $\mathcal{T}_m$	-0.530	-0.550	-0.538	-0.539	-0.589	-0.595	-0.427	-0.433	-0.590	-0.608
	sJS $\mathcal{T}_m$	-0.533	-0.556	-0.530	-0.538	-0.588	-0.595	-0.423	-0.433	-0.595	-0.605
	KL $\mathcal{T}_m$	-0.441	-0.497	-0.430	-0.459	-0.475	-0.514	-0.376	-0.405	-0.442	-0.504
	JS $\mathcal{T}_c$	-0.397	-0.406	-0.507	-0.501	-0.610	-0.607	-0.404	-0.409	-0.563	-0.578
	sJS $\mathcal{T}_c$	-0.397	-0.406	-0.504	-0.501	-0.608	-0.607	-0.406	-0.409	-0.565	-0.577
	KL $\mathcal{T}_c$	-0.075	-0.063	-0.009	0.113	0.135	0.130	0.093	0.114	-0.124	-0.091

Table A.72: Correlation coefficients in terms of Kendall of SummTriver with Responsiveness on TAC 2008 dataset, using 600 and 900 summary per corpus

A.3.3 Correlation of SummTriver with Pyramid on TAC 2009

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
Pyramid M3	JS mul	-0.461	-0.475	-0.447	-0.470	-0.506	-0.526	-0.389	-0.405	-0.427	-0.450
	sJS mul	-0.452	-0.468	-0.433	-0.459	-0.488	-0.511	-0.376	-0.393	-0.403	-0.427
	KL mul	-0.227	-0.286	-0.246	-0.316	-0.323	-0.371	-0.261	-0.301	-0.273	-0.319
	JS com	-0.295	-0.307	-0.401	-0.416	-0.461	-0.477	-0.385	-0.392	-0.410	-0.423
	sJS com	-0.295	-0.307	-0.400	-0.414	-0.460	-0.475	-0.383	-0.390	-0.405	-0.417
	KL com	-0.039	0.059	-0.100	-0.084	-0.176	-0.138	-0.089	-0.056	0.042	0.059
Pyramid M4	JS mul	-0.459	-0.473	-0.447	-0.469	-0.504	-0.524	-0.388	-0.404	-0.423	-0.445
	sJS mul	-0.449	-0.465	-0.432	-0.458	-0.487	-0.509	-0.374	-0.391	-0.398	-0.422
	KL mul	-0.225	-0.283	-0.246	-0.316	-0.323	-0.371	-0.258	-0.299	-0.270	-0.315
	JS com	-0.293	-0.305	-0.400	-0.415	-0.459	-0.475	-0.384	-0.390	-0.406	-0.419
	sJS com	-0.293	-0.305	-0.399	-0.414	-0.457	-0.473	-0.382	-0.388	-0.400	-0.413
	KL com	-0.041	0.058	-0.101	-0.085	-0.180	-0.142	-0.089	-0.057	0.042	0.058

Table A.73: Correlation coefficients in terms of Pearson of SummTriver with Pyramid on TAC 2009 dataset, using 900 summaries per corpus

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
Pyramid M3	JS mul	-0.745	-0.731	-0.717	-0.715	-0.759	-0.755	-0.680	-0.669	-0.719	-0.720
	sJS mul	-0.755	-0.740	-0.711	-0.713	-0.754	-0.751	-0.674	-0.670	-0.718	-0.712
	KL mul	-0.585	-0.597	-0.522	-0.591	-0.633	-0.681	-0.575	-0.628	-0.608	-0.621
	JS com	-0.600	-0.604	-0.676	-0.682	-0.718	-0.718	-0.690	-0.686	-0.706	-0.702
	sJS com	-0.600	-0.604	-0.676	-0.681	-0.716	-0.717	-0.689	-0.686	-0.706	-0.702
	KL com	-0.024	0.165	-0.118	-0.063	-0.104	-0.062	-0.052	0.015	0.069	0.087
Pyramid M4	JS mul	-0.745	-0.732	-0.718	-0.716	-0.757	-0.754	-0.682	-0.672	-0.716	-0.718
	sJS mul	-0.754	-0.740	-0.712	-0.715	-0.752	-0.749	-0.676	-0.673	-0.716	-0.709
	KL mul	-0.584	-0.597	-0.524	-0.593	-0.633	-0.681	-0.577	-0.630	-0.607	-0.620
	JS com	-0.601	-0.605	-0.679	-0.685	-0.714	-0.714	-0.691	-0.688	-0.704	-0.701
	sJS com	-0.601	-0.605	-0.679	-0.684	-0.712	-0.713	-0.690	-0.688	-0.704	-0.700
	KL com	-0.020	0.167	-0.114	-0.058	-0.101	-0.059	-0.058	0.010	0.066	0.085

Table A.74: Correlation coefficients in terms of Spearman of SummTriver with Pyramid on TAC 2009 dataset, using 900 summaries per corpus

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
Pyramid M3	JS mul	-0.603	-0.591	-0.589	-0.584	-0.626	-0.623	-0.562	-0.547	-0.608	-0.599
	sJS mul	-0.611	-0.601	-0.588	-0.589	-0.626	-0.620	-0.553	-0.552	-0.604	-0.592
	KL mul	-0.442	-0.461	-0.455	-0.492	-0.552	-0.558	-0.467	-0.507	-0.523	-0.525
	JS com	-0.446	-0.453	-0.553	-0.554	-0.585	-0.582	-0.566	-0.557	-0.588	-0.587
	sJS com	-0.446	-0.452	-0.553	-0.552	-0.585	-0.581	-0.566	-0.558	-0.585	-0.585
	KL com	-0.022	0.119	-0.076	-0.040	-0.075	-0.040	-0.038	0.015	0.044	0.065
Pyramid M4	JS mul	-0.603	-0.591	-0.589	-0.587	-0.626	-0.623	-0.561	-0.549	-0.605	-0.599
	sJS mul	-0.611	-0.601	-0.591	-0.592	-0.626	-0.620	-0.552	-0.553	-0.601	-0.592
	KL mul	-0.440	-0.461	-0.460	-0.498	-0.552	-0.558	-0.468	-0.508	-0.523	-0.525
	JS com	-0.449	-0.453	-0.556	-0.557	-0.582	-0.580	-0.565	-0.553	-0.588	-0.587
	sJS com	-0.449	-0.455	-0.556	-0.554	-0.582	-0.578	-0.565	-0.554	-0.585	-0.585
	KL com	-0.017	0.122	-0.079	-0.037	-0.072	-0.037	-0.040	0.011	0.044	0.065

Table A.75: Correlation coefficients in terms of Kendall of SummTriver with Pyramid on TAC 2009 dataset, using 900 summaries per corpus

A.3.4 Correlation of SummTriver with Responsiveness on TAC 2009

	Set Size	2 Summaries		5 Summaries		10 Summaries		15 Summaries		30 Summaries	
	Methods	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4	Bigram	SU4
Pearson	JS mul	-0.495	-0.501	-0.436	-0.449	-0.637	-0.650	-0.511	-0.523	-0.416	-0.427
	sJS mul	-0.493	-0.501	-0.422	-0.437	-0.620	-0.636	-0.499	-0.513	-0.396	-0.407
	KL mul	-0.320	-0.363	-0.252	-0.305	-0.473	-0.518	-0.382	-0.426	-0.294	-0.323
	JS com	-0.363	-0.371	-0.390	-0.396	-0.609	-0.619	-0.524	-0.530	-0.384	-0.389
	sJS com	-0.363	-0.371	-0.388	-0.394	-0.607	-0.618	-0.523	-0.528	-0.379	-0.384
	KL com	0.025	0.078	-0.007	0.017	-0.034	0.014	-0.043	0.007	-0.031	-0.020
Spearman	JS mul	-0.541	-0.515	-0.534	-0.512	-0.748	-0.744	-0.590	-0.574	-0.553	-0.547
	sJS mul	-0.558	-0.528	-0.530	-0.513	-0.742	-0.739	-0.588	-0.575	-0.553	-0.539
	KL mul	-0.483	-0.455	-0.407	-0.449	-0.622	-0.683	-0.525	-0.569	-0.458	-0.450
	JS com	-0.481	-0.461	-0.517	-0.506	-0.712	-0.710	-0.624	-0.618	-0.531	-0.529
	sJS com	-0.481	-0.461	-0.517	-0.506	-0.712	-0.709	-0.623	-0.617	-0.531	-0.529
	KL com	0.073	0.188	0.007	0.057	-0.033	0.007	-0.011	0.063	-0.039	-0.030
Kendall	JS mul	-0.390	-0.370	-0.412	-0.387	-0.587	-0.587	-0.442	-0.442	-0.432	-0.418
	sJS mul	-0.404	-0.381	-0.413	-0.393	-0.585	-0.585	-0.441	-0.444	-0.428	-0.411
	KL mul	-0.370	-0.343	-0.340	-0.365	-0.527	-0.550	-0.390	-0.422	-0.356	-0.349
	JS com	-0.336	-0.316	-0.397	-0.385	-0.566	-0.563	-0.483	-0.476	-0.418	-0.416
	sJS com	-0.336	-0.317	-0.397	-0.385	-0.566	-0.562	-0.483	-0.475	-0.415	-0.415
	KL com	0.045	0.119	0.007	0.044	-0.030	0.005	-0.012	0.042	-0.034	-0.020

Table A.76: Correlation coefficients in terms of Pearson, Spearman and Kendall of SummTriver with Responsiveness on TAC 2009 dataset, using 900 summaries per corpus

A.4 FRESA

A.4.1 Correlation of FRESA with Pyramid and Responsiveness on TAC 2008

	Pyramid						Responsiveness		
	Average score with 3 reference summaries			Average score with 4 reference summaries			-		
	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
FRESA_1	-0.487	-0.638	-0.537	-0.486	-0.636	-0.537	-0.385	-0.498	-0.371
FRESA_2	0.474	-0.062	-0.064	0.475	-0.063	-0.064	0.523	0.076	0.034
FRESA_3	0.539	0.241	0.162	0.540	0.238	0.162	0.593	0.362	0.250
FRESA_4	0.544	0.257	0.168	0.544	0.255	0.168	0.596	0.416	0.296
FRESA_M	0.464	-0.090	-0.090	0.464	-0.090	-0.090	0.523	0.081	0.040

Table A.77: Correlation coefficients, in terms of Pearson, Spearman and Kendall, of FRESA with Pyramid and Responsiveness, on TAC 2008 dataset

A.4.2 Correlation of FRESA with Pyramid and Responsiveness on TAC 2009

	Pyramid						Responsiveness		
	Average score with 3 reference summaries			Average score with 4 reference summaries			-		
	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall	Pearson	Spearman	Kendall
FRESA_1	-0.610	-0.650	-0.491	-0.609	-0.652	-0.493	-0.594	-0.565	-0.410
FRESA_2	-0.630	-0.046	-0.026	-0.630	-0.048	-0.027	-0.385	0.074	0.063
FRESA_3	-0.556	0.055	0.056	-0.556	0.054	0.055	-0.298	0.180	0.147
FRESA_4	-0.516	0.189	0.142	-0.516	0.187	0.141	-0.217	0.369	0.278
FRESA_M	-0.643	-0.066	-0.041	-0.643	-0.069	-0.042	-0.373	0.090	0.062

Table A.78: Correlation coefficients, in terms of Pearson, Spearman and Kendall, of FRESA with Pyramid and Responsiveness, on TAC 2009 dataset

Appendix B

Résumé en français

Les dernières statistiques faites par l'IDC (International Data Corporation)[1] montrent que le volume d'information en exabytes dans le domaine médical a augmenté de plus de 1400% entre les années 2013 et 2020. Cette croissance monstrueuse fait que des sites tel que "PubMed" (for Biotechnology Information, 2018) de "MEDLINE" (Solutions, 2021) et "Dimensions" contiennent à présent des millions d'articles médicaux portant sur des sujets variés. Cependant, et afin de suivre le rapide progrès dans le domaine médical, les chercheurs et les médecins ont besoin d'accéder aux informations pertinentes le plus rapidement possible.

Grâce à l'intelligence artificielle et les avancements dans le traitement automatique du langage naturel, le domaine du résumé automatique de textes a émergé pour le but de proposer des solutions efficaces afin de transformer un ou plusieurs textes longs en un résumé de petite taille concentrant leur information la plus utile.

Les premiers travaux dans le domaine du résumé automatique étaient extractifs, où les phrases les plus pertinentes du texte sont copiées et concaténées afin de construire le résumé. Avec l'apparition de l'apprentissage profond, le résumé automatique est basé désormais sur des approches abstractives, où le système reformule le texte en un résumé qui ne contient pas forcément des mots du texte original.

Malgré l'évolution dans le domaine du résumé automatique, il est nécessaire d'évaluer automatiquement la qualité des résumés générés afin de pouvoir comparer et améliorer les différentes approches de l'état de l'art. Ceci dit que le domaine d'évaluation automatique des résumés est aussi important pour le fait que l'évaluation manuelle est coûteuse en termes d'argent et de temps, même si elle constitue la meilleure référence d'évaluation.

Il existe deux types d'approches automatiques d'évaluation de résumé : celles qui nécessitent une intervention humaine (telles que ROUGE (Lin, 2004) et SERA (Cohan and Goharian, 2016)), et celles qui ne la nécessitent pas (telles que SummTriver (Cabrera-Diego and Torres-Moreno, 2018) et FRESA (Torres-Moreno et al., 2010)). Les dernières approches ont l'avantage de fonctionner sans avoir besoin d'un résumé

[1] https://www.idc.com/

de référence, mais elles ont jusqu'à présent une faible corrélation avec les méthodes d'évaluation manuelles.

Dans cette thèse, nous nous focalisons sur le résumé automatique abstractif des textes médicaux longs, ainsi que l'évaluation automatique des résumés appartenant au domaine général. Pour la première problématique, nous proposons une amélioration de l'architecture originale des réseaux de neurones de type Transformers. Notre méthode (appelée *HazPi*) consiste à augmenter le nombre d'encodeurs du modèle en découpant l'entrée entre eux afin de concentrer l'attention du modèle sur des sous parties du texte (Multi-encoder Transformer). En plus, notre méthode favorise l'apprentissage progressif en présentant les résumés au décodeur partie par partie jusqu'à la consommation de toute la séquence (End-chunk Task Training). Nous menons des expérimentations sans et avec pré-entraînement du modèle sur des datasets médicales et les résultats obtenus sont encourageants en comparant *HazPi* avec des méthodes compétitives de l'état de l'art.

Pour la deuxième problématique, nous présentons wikiSERA, une amélioration de la méthode SERA pour l'évaluation automatique des résumés biomédicaux en se basant sur l'intervention humaine. SERA est basée sur une analyse de la pertinence de contenu entre un résumé candidat et un ensemble de résumés de référence à l'aide d'un moteur de recherche qui compare les résultats de recherche dans un ensemble de documents qui constituent l'index, avec comme requêtes en entrée d'une part les résumés de référence et d'autre part les résumés automatiques.

Nous proposons de redéfinir la reformulation des requêtes dans SERA en utilisant les parties du discours et d'utiliser comme index des documents extraits de l'encyclopédie Wikipédia, afin de s'abstraire du domaine biomédical vers le domaine général. Notre méthode, wikiSERA, est un système sous licence libre disponible pour la communauté et prêt pour évaluer des résumés du domaine général. Les résultats obtenus montrent que wikiSERA améliore les résultats de SERA dans le domaine général et parfois même dépasse les scores obtenus par ROUGE.

De plus, nous menons des expérimentations approfondies sur les performances de SERA et wikiSERA sur les datasets TAC 2008, TAC 2009 et CNNDM afin d'établir une comparaison complète sous plusieurs contraintes telles que l'impact de la taille de l'index et la performance de chaque résumateur humain.

Bibliography

R.M. Aliguliyev. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications*, 36:7764–7772, 2009. (Cited on pages 27, 46 and 48.)

Abdelkrime Aries, Djamel Eddine Zegour, and Walid-Khaled Hidouci. Automatic text summarization: What has been done and what has to be done. *CoRR*, abs/1904.00688, 2019. (Cited on page 65.)

arXiv. arxiv.org, 2021. URL `https://arxiv.org/`. (Cited on pages 19 and 50.)

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. (Cited on page 48.)

Forrest Sheng Bao, Hebi Li, Ge Luo, Cen Chen, Yinfei Yang, and Minghui Qiu. End-to-end semantics-based summary quality assessment for single-document summarization. *CoRR*, abs/2005.06377, 2020. (Cited on pages 62 and 63.)

Nadira Begun, Mohamed A. Fattah, and Fuji Ren. Automatic text summarization using support vector machine. *International Journal of Innovative Computing, Information and Control*, 5:1987–1996, 07 2009. (Cited on pages 14 and 28.)

Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. In *EMNLP/IJCNLP*, 2019. (Cited on page 47.)

Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise reduction in speech processing*, pages 1–4. Springer, 2009. (Cited on page 73.)

Manik Bhandari, Pranav Narayan Gour, Atabak Ashfaq, Pengfei Liu, and Graham Neubig. Re-evaluating evaluation in text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9347–9359, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.751. URL `https://www.aclweb.org/anthology/2020.emnlp-main.751`. (Cited on pages 63, 67, 69, 71, 72, 75, 80, 89 and 115.)

Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pascal Vincent. Audio chord recognition with recurrent neural networks. *ISMIR 2013*, pages 335–340, 2013. (Cited on pages 33 and 44.)

Luis Adrián Cabrera-Diego and Juan-Manuel Torres-Moreno. Summtriver: A new trivergent model to evaluate summaries automatically without human references. *Data Knowl. Eng.*, 113:184–197, 2018. (Cited on pages 6, 51, 59, 62, 65, 66, 69, 74, 75 and 195.)

Luis Adrián Cabrera-Diego, Juan-Manuel Torres-Moreno, and Barthélémy Durette. Evaluating multiple summaries without human models: A first experiment with a trivergent model. In *International conference on applications of natural language to information systems*, pages 91–101. Springer, 2016. (Cited on page 66.)

Jaime Carbonell and Jade Stewart. The use of mmr, diversity-based reranking for reordering documents and producing summaries. *SIGIR Forum (ACM Special Interest Group on Information Retrieval)*, 06 1998. doi: 10.1145/290941.291025. (Cited on page 49.)

Eddy Cardinaels, Stephan Hollander, and Brian J. White. Automatic summarization of earnings releases: attributes and effects on investors' judgments. *Review of Accounting Studies*, 24(3):860–890, 2018. (Cited on page 16.)

Jean Carletta. Assessing agreement on classification tasks: The kappa statistic. *Comput. Linguistics*, 22(2):249–254, 1996. (Cited on page 52.)

Matt Chaput. Whoosh. `https://whoosh.readthedocs.io/en/latest/intro.html`, 2007-2012. Accessed: 2019-06-02. (Cited on page 74.)

Kushal Chauhan. Unsupervised text summarization using sentence embeddings. `https://medium.com/jatana/unsupervised-text-summarization-using-sentence-embeddings-adb15ce83db1`, 2018. (Cited on page 16.)

Francine R. Chen and Margaret Withgott. The use of emphasis to automatically summarize a spoken discourse. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 229–232, 1992. (Cited on pages 13 and 24.)

Yen-Chun Chen and Mohit Bansal. Fast abstractive summarization with reinforce-selected sentence rewriting. *ArXiv*, abs/1805.11080, 2018. (Cited on page 72.)

J. Cheng and M. Lapata. Neural summarization by extracting sentences and words. *P16-1046*, Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers):484–494, 2016. (Cited on pages 14, 15, 34, 47 and 48.)

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representation using rnn encoder-decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, 2014. (Cited on pages 14, 33, 47 and 91.)

François Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015. (Cited on page 98.)

Sumit Chopra, Michael Auli, and Alexander M. Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98, San Diego, California, 06 2016. (Cited on pages 48, 51 and 65.)

Arman Cohan and Nazli Goharian. Revisiting summarization evaluation for scientific articles. *CoRR*, abs/1604.00400, 2016. (Cited on pages 5, 6, 8, 51, 56, 63, 66, 67, 68, 69, 84, 86, 87, 99, 106 and 195.)

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2097. URL `https://www.aclweb.org/anthology/N18-2097`. (Cited on pages 4, 5, 7, 9, 14, 17, 19, 48, 50, 67, 92, 94, 96, 97, 98, 101, 102, 112 and 114.)

John M. Conroy and Dianne P. O'leary. Text summarization via hidden markov models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, page 406–407, New York, NY, USA, 2001. Association for Computing Machinery. ISBN 1581133316. doi: 10.1145/383952.384042. URL https://doi.org/10.1145/383952.384042. (Cited on pages 13, 24, 45 and 46.)

Linguistic Data Consortium. Aquaint-2 information-retrieval text research collection. https://catalog.ldc.upenn.edu/LDC2008T25, 2008. (Cited on pages 17 and 87.)

Franck Dernoncourt and Ji Young Lee. PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan, November 2017. Asian Federation of Natural Language Processing. URL https://www.aclweb.org/anthology/I17-2052. (Cited on pages 14, 19 and 47.)

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423. (Cited on pages 4, 14, 38, 39, 47, 49, 68, 103, 112 and 113.)

Li Dong, Nan Yang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 05 2019. (Cited on pages 14 and 72.)

Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. BanditSum: Extractive summarization as a contextual bandit. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3739–3748, Brussels, Belgium, October-November 2018.

Association for Computational Linguistics. doi: 10.18653/v1/D18-1409. URL https://www.aclweb.org/anthology/D18-1409. (Cited on page 72.)

H.P. Edmundson. New methods in automatic extracting. *Journal of the ACM*, 16(2):264–285, 1969. (Cited on page 15.)

G. Erkan and D.G. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, 2004. (Cited on pages 21 and 46.)

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1102. URL https://www.aclweb.org/anthology/P19-1102. (Cited on pages 14, 18, 49, 50, 51, 91 and 93.)

National Center for Biotechnology Information. Pubmed. https://www.ncbi.nlm.nih.gov/pubmed/, 2018. Accessed: 2018-12-14. (Cited on pages 1, 8, 112 and 195.)

Yanjun Gao, Chen Sun, and Rebecca J. Passonneau. Automated pyramid summarization evaluation. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 404–418, Hong Kong, China, November 2019. Association for Computational Linguistics. (Cited on page 54.)

Jorge García Flores, Olivier Ferret, and Gaël de Chalendar. Summarizing through sense concentration and contextual exploration rules: the CHORAL system at TAC 2009. In *Proceedings of the Second Text Analysis Conference, TAC 2009, Gaithersburg, Maryland, USA, November 16-17, 2009*, 2009. (Cited on page 7.)

Sebastian Gehrmann, Yuntian Deng, and Alexander M Rush. Bottom-up abstractive summarization. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4098–4109. Association for Computational Linguistics, 2018. (Cited on pages 72 and 97.)

George Giannakopoulos and V. Karkaletsis. Autosummeng and memog in evaluating guided summaries. *Theory and Applications of Categories*, 2011. (Cited on page 66.)

Dan Gillick and Yang Liu. Non-expert evaluation of summarization systems is risky. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 148–151, 07 2010. (Cited on page 115.)

David Graff. *The AQUAINT corpus of English news text:[content copyright] Portions© 1998-2000 New York Times, Inc.,© 1998-2000 Associated Press, Inc.,© 1996-2000 Xinhua News Service.* Linguistic Data Consortium, 2002. (Cited on pages 9 and 71.)

Alex Graves. Sequence transduction with recurrent neural networks. arXiv:1211.3711, 2012. (Cited on pages 33 and 44.)

Max Grusky, Mor Naaman, and Yoav Artzi. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1065. URL `https://www.aclweb.org/anthology/N18-1065`. (Cited on pages 18 and 50.)

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, Cambridge, MA, USA, 2015. MIT Press. (Cited on pages 14, 18, 47, 50 and 75.)

Andrew Hoang, Antoine Bosselut, Asly Celikyilmaz, and Yejing Choi. Efficient adaptation of pretrained transformers for abstractive summarization. *ArXiv*, June 2019. URL `https://arxiv.org/pdf/1906.00138.pdf`. (Cited on pages 7, 49, 92, 94, 95, 109 and 116.)

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. (Cited on pages 14, 31, 47 and 91.)

The White House. Covid-19 open research dataset challenge (cord-19), 2020. (Cited on pages 8, 97 and 112.)

Yanping Huang, Youlong Cheng, Ankur Bapna, Orhan Firat, Dehao Chen, Mia Xu Chen, HyoukJoong Lee, Jiquan Ngiam, Quoc V. Le, Yonghui Wu, and Zhifeng Chen. Gpipe: Efficient training of giant neural networks using pipeline parallelism. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 103–112, 2019. URL `https://proceedings.neurips.cc/paper/2019/hash/093f65e080a295f8076b1c5722a46aa2-Abstract.html`. (Cited on pages 104 and 113.)

H. Jing and K.R. McKeown. The decomposition of human-written summary sentences. *Proceedings of SIGIR-99*, pages 129–136, 1999. (Cited on pages 13, 24, 45, 46 and 91.)

Karen Sparck Jones and Julia R. Galliers. *Evaluating Natural Language Processing Systems: An Analysis and Review.* Springer-Verlag, Berlin, Heidelberg, 1996. ISBN 3540613099. (Cited on page 3.)

V. Kecman. *Support Vector Machines – An Introduction. In: Wang L. (eds) Support Vector Machines: Theory and Applications*, volume 177. Springer, Berlin, Heidelberg, 2005. (Cited on page 28.)

Chris Kedzie, Kathleen McKeown, and Hal Daumé III. Content selection in deep learning models of summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1818–1828, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1208. URL `https://www.aclweb.org/anthology/D18-1208`. (Cited on page 72.)

Maurice G Kendall. The treatment of ties in ranking problems. *Biometrika*, pages 239–251, 1945. (Cited on page 74.)

Nitish Shirish Keskar, Bryan McCann, Lav Varshney, Caiming Xiong, and Richard Socher. CTRL - A Conditional Transformer Language Model for Controllable Generation. *arXiv preprint arXiv:1909.05858*, 2019. (Cited on pages 104 and 113.)

Virapat Kieuvongngam, Bowen Tan, and Yiming Niu. Automatic text summarization of COVID-19 medical research articles using BERT and GPT-2.

CoRR, abs/2006.01997, 2020. URL https://arxiv.org/abs/2006.01997. (Cited on pages 9, 67 and 87.)

Byeongchang Kim, Hyunwoo Kim, and Gunhee Kim. Abstractive summarization of Reddit posts with multi-level memory networks. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2519–2531, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1260. URL https://www.aclweb.org/anthology/N19-1260. (Cited on pages 19, 49, 50 and 91.)

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980. (Cited on page 98.)

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3294–3302, 2015. (Cited on page 58.)

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. (Cited on page 43.)

Bryan Klimt and Yiming Yang. The enron corpus: A new dataset for email classification research. In *Springer-Verlag*, Berlin, Heidelberg, 2004. ISBN 3540231056. doi: 10.1007/978-3-540-30115-8_22. URL https://doi.org/10.1007/978-3-540-30115-8_22. (Cited on page 19.)

K. Knight and D. Marcu. Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, 139: 91–107, 2002. (Cited on pages 13, 23, 45 and 46.)

Stephen Kokoska and Daniel Zwillinger. *CRC standard probability and statistics tables and formulae.* Crc Press, 2000. (Cited on page 73.)

Anastassia Kornilova and Vladimir Eidelman. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-5406. URL `https://www.aclweb.org/anthology/D19-5406`. (Cited on pages 18 and 50.)

Simeon Kostadino. Understanding encoder-decoder sequence to sequence model. `https://towardsdatascience.com/understanding-encoder-decoder-sequence-to-sequence-model-679e04af4346`, 2019. (Cited on page 29.)

Mahnaz Koupaee and William Wang. Wikihow: A large scale text summarization dataset. In *arXiv:1810.09305*, 10 2018. (Cited on pages 19 and 50.)

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.750. URL `https://www.aclweb.org/anthology/2020.emnlp-main.750`. (Cited on page 115.)

Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 66–75. Association for Computational Linguistics, 2018. doi: 10.18653/v1/P18-1007. URL `https://www.aclweb.org/anthology/P18-1007/`. (Cited on page 41.)

Y.J. Kumar and N. Salim. Automatic multi document summarization approaches. *Journal of Computer Science*, 8(1):133–140, 2012. (Cited on page 16.)

Y.J. Kumar, O.S. Goh, H. Basiron, N.H. Choon, and P. Suppiah. A review on automatic text summarization approaches. *Journal of Computer Science*, 12(4): 178–190, 2016. (Cited on pages 15, 21, 27 and 47.)

J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. *In Proceedings of the 18th ACM-SIGIR Conference*, pages 68–73, 1995. (Cited on pages 15 and 27.)

Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. From word embeddings to document distances. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 957–966. JMLR.org, 2015. (Cited on page 59.)

Alon Lavie and Abhaya Agarwal. METEOR: an automatic metric for MT evaluation with high levels of correlation with human judgments. In Chris Callison-Burch, Philipp Koehn, Cameron S. Fordyce, and Christof Monz, editors, *Proceedings of the Second Workshop on Statistical Machine Translation, WMT@ACL 2007, Prague, Czech Republic, June 23, 2007*, pages 228–231. Association for Computational Linguistics, 2007. URL https://www.aclweb.org/anthology/W07-0734/. (Cited on page 58.)

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://www.aclweb.org/anthology/2020.acl-main.703. (Cited on pages 38, 42, 49, 50, 72 and 91.)

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain, July 2004. (Cited on pages 5, 6, 8, 51, 54, 58, 62, 65, 66, 67, 69, 74, 99, 114 and 195.)

Chin-Yew Lin and Eduard Hovy. Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 45–51, Phildadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1118162.1118168. URL https://www.aclweb.org/anthology/W02-0406. (Cited on pages 52 and 65.)

Chin-Yew Lin, Guihong Cao, Jianfeng Gao, and Jian-Yun Nie. An information-theoretic approach to automatic evaluation of summaries. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 463–470, New York City, USA, June 2006. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/N06-1059`. (Cited on pages 61 and 70.)

J. Lin. Summarization. *Encyclopedia of Database Systems*, 2009. (Cited on page 15.)

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1112. URL `https://www.aclweb.org/anthology/N19-1112`. (Cited on pages 104, 109 and 113.)

Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. *CoRR*, abs/1801.10198, 2018. (Cited on pages 18, 43, 49, 50 and 116.)

Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. In *EMNLP/IJCNLP*, 2019a. (Cited on pages 47, 49, 50, 51, 65 and 72.)

Yang Liu and Mirella Lapata. Hierarchical transformers for multi-document summarization. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5070–5081. Association for Computational Linguistics, 2019b. (Cited on page 43.)

Edward Loper and Steven Bird. Nltk: The natural language toolkit. In *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*, 2002. (Cited on page 74.)

A. Louis and A. Nenkova. Automatically evaluating content selection in summarization without human models. *Conference on Empirical Methods in Natural Language Processing*, pages 306–314, 2009. (Cited on page 59.)

Mingyi Lu and Xiaomeng Jin. Clinicalbertsum: Rct summarization by using clinical bert embeddings. In *Stanford CS224N Natural Language Processing with Deep Learning*, 2020. (Cited on pages 8, 47 and 66.)

Yao Lu, Yue Dong, and Laurent Charlin. Multi-xscience: A large-scale dataset for extreme multi-documentsummarization of scientific articles. In *ACL 2019, 57th Annual Meeting of the Association for Computational Linguistics*, 2019. (Cited on page 19.)

H.P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research Development*, 2:159–165, 1958. (Cited on pages 13, 15, 20, 21, 45, 63 and 91.)

Inderjeet Mani. *Automatic Summarization.* John Benjamins Publishing, 2001. (Cited on page 2.)

C.D. Manning and H. Schutze. *Foundations of Statistical Natural Language Processing.* Cambridge, Mass. : MIT Press, 1999. (Cited on pages 22, 23, 24, 26 and 27.)

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60. The Association for Computer Linguistics, 2014. (Cited on page 54.)

Rada Mihalcea and Paul Tarau. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411. Association for Computational Linguistics, 2004. (Cited on page 46.)

Tomás Mikolov, Kai Chen, Greg Corrado, and Dean. Jeffrey. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL `http://arxiv.org/abs/1301.3781`. (Cited on page 33.)

Jeff Mitchell and Mirella Lapata. Vector-based models of semantic composition. In *proceedings of ACL-08: HLT*, pages 236–244, 2008. (Cited on page 56.)

Ruslan Mitkov. *The Oxford handbook of computational linguistics*. Oxford University Press, 2004. (Cited on pages 5 and 71.)

Ramesh Nallapati, Bowen Zhou, Cicero Dos Santos, Caglar Gulcehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany, 08 2016. (Cited on pages 14, 15, 18, 48, 50, 51, 65, 75 and 91.)

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. *Thirty-First AAAI Conference on Artificial Intelligence*, 2017. (Cited on pages 14, 34, 47 and 91.)

Courtney Napoles, Matthew Gormley, and Benjamin Van Durme. Annotated gigaword. In *Association for Computational Linguistics*, AKBC-WEKEX '12, page 95–100, USA, 2012. (Cited on pages 17 and 51.)

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018a. (Cited on pages 18, 49, 50 and 91.)

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1747–1759, New Orleans, Louisiana, June 2018b. Association for Computational Linguistics. doi: 10.18653/v1/N18-1158. URL https://www.aclweb.org/anthology/N18-1158. (Cited on page 72.)

nature. How a torrent of covid science changed research publishing — in seven charts. https://www.nature.com/articles/d41586-020-03564-y, 2020. (Cited on page 1.)

Ani Nenkova. Summarization evaluation for text and speech: Issues and approaches. *ICSLP*, pages 1527–1530, 2006. (Cited on pages 51, 52 and 53.)

Ani Nenkova and Rebecca J. Passonneau. Evaluating content selection in summarization: The pyramid method. In *HLT-NAACL*, pages 145–152, 2004. URL `http://acl.ldc.upenn.edu/hlt-naacl2004/main/pdf/91Paper.pdf`. (Cited on pages 51, 53, 54, 66, 67, 68 and 72.)

Ani Nenkova and Lucy Vanderwende. The impact of frequency on summarization. Technical report, Microsoft Research, 2005. (Cited on pages 46 and 48.)

Jun-Ping Ng and Viktoria Abrecht. Better summarization evaluation with word embeddings for ROUGE. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1925–1930. The Association for Computational Linguistics, 2015. (Cited on page 56.)

NIST. Document understanding conferences. `https://www-nlpir.nist.gov/projects/duc/index.html`, 2014. (Cited on pages 16, 48 and 51.)

NIST. Text retrieval conference. `https://trec.nist.gov/`, 2020. (Cited on page 17.)

Paul Over, Hoa Dang, and Donna Harman. Duc in context. *Inf. Process. Manage.*, 43(6):1506–1520, November 2007. ISSN 0306-4573. doi: 10.1016/j.ipm.2007.01.019. URL `https://doi.org/10.1016/j.ipm.2007.01.019`. (Cited on pages 17 and 48.)

Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. An assessment of the accuracy of automatic evaluation in summarization. In John M. Conroy, Hoa Trang Dang, Ani Nenkova, and Karolina Owczarzak, editors, *Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization@NACCL-HLT 2012, Montrèal, Canada, June 2012, 2012*, pages 1–9. Association for Computational Linguistics, 2012. (Cited on page 52.)

K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the*

Association for Computational Linguistics (ACL), pages 311–318, 2002. (Cited on pages 58 and 69.)

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304, 2017. URL http://arxiv.org/abs/1705.04304. (Cited on pages 48, 50 and 51.)

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011. (Cited on page 74.)

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, page 2227–2237, New Orleans, Louisiana, 2018. (Cited on page 33.)

D.R. Radev, D. Tam, and G. Erkan. Single-document and multi-document summary evaluation using relative utility. *Poster session, CIKM'03*, 2003. (Cited on page 52.)

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. In *arxiv*, 2018. (Cited on pages 42 and 49.)

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. *Technical report, OpenAi*, 2019. URL https://openai.com/blog/better-language-models/. (Cited on pages 104 and 113.)

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html. (Cited on pages 4, 14, 18, 38, 40, 41, 72, 97, 103, 109, 112 and 113.)

A. F. R. Rahman, H. Alam, R. Hartono, and K. Ariyoshi. Automatic summarization of web content to smaller display devices. In *Proceedings of Sixth International Conference on Document Analysis and Recognition*, pages 1064–1068, 2001. doi: 10.1109/ICDAR.2001.953949. (Cited on pages 13 and 23.)

N. Ramanujam and M. Kaliappan. An automatic multidocument text summarization approach based on naïve bayesian classifier using timestamp strategy. In *TheScientificWorldJournal*, 2016. (Cited on pages 14, 27 and 46.)

Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vis.*, 40(2):99–121, 2000. (Cited on pages 59 and 70.)

A.M. Rush, S. Chopra, and J. Weston. A neural attention model for sentence summarization. *Association for Computational Linguistics*, Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing: 379–389, 2015. (Cited on pages 15, 17, 48, 51 and 91.)

S. Russell and P. Norving. *Artificial Intelligence (A Modern Approach)*. Prentice–Hall Hispanoamericana, 2010. (Cited on page 24.)

Evan Sandhaus. *The New York Times Annotated Corpus*. LDC corpora. Linguistic Data Consortium, 2008. URL `https://catalog.ldc.upenn.edu/LDC2008T19`. (Cited on pages 17, 48 and 51.)

K. Sarkar. Using domain knowledge for text summarization in medical domain. *Int. J. of Recent Trends in Engineering and Technology*, 1(1):200–205, 2009. (Cited on page 16.)

F. Schilder and R. Kondadadi. Fastsum: Fast and accurate query-based multi-document summarization. In *ACL*, 2008. (Cited on pages 14, 28, 46 and 48.)

A. See, P.J. Liu, and C.D. Manning. Get to the point: Summarization with pointer-generator networks. *Trans. Amer. Math. Soc.*, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers):1073–1083, 2017. (Cited on pages 5, 7, 14, 48, 49, 50, 65, 72, 91, 97, 98 and 105.)

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL `https://www.aclweb.org/anthology/P16-1162`. (Cited on page 41.)

F. Shaikh. Essentials of deep learning – sequence to sequence modelling with attention (using python). `https://www.analyticsvidhya.com/blog/2018/03/essentials-of-deep-learning-sequence-to-sequence-modelling-with-attention-part`
2018. (Cited on page 14.)

Ori Shapira, David Gabay, Yang Gao, Hadar Ronen, Ramakanth Pasunuru, Mohit Bansal, Yael Amsterdamer, and Ido Dagan. Crowdsourcing lightweight pyramids for manual summary evaluation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 682–687, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1072. URL `https://www.aclweb.org/anthology/N19-1072`. (Cited on pages 54 and 73.)

Eva Sharma, Chen Li, and Lu Wang. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1212. URL `https://www.aclweb.org/anthology/P19-1212`. (Cited on pages 19 and 50.)

K.M. ShivaKumar and R. Soumya. Text summarization using clustering technique and svm technique. *International Journal of Applied Engineering Research*, 10: 25511–25519, 2015. (Cited on pages 14, 27 and 46.)

Sakai Shuichi, Togasaki Mitsunori, and Yamazaki Koichi. A note on greedy algorithms for the maximum weighted independent set problem. *Discret. Appl. Math.*, 126(2-3):313–322, 2003. doi: 10.1016/S0166-218X(02)00205-6. URL `https://doi.org/10.1016/S0166-218X(02)00205-6`. (Cited on page 54.)

A. Sinha, A. Yadav, and A. Gahlot. Extractive text summarization using neural networks. *CoRR*, abs/1802.10137, 2018. URL http://arxiv.org/abs/1802.10137. (Cited on pages 47 and 48.)

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Bo-June (Paul) Eide, Darrin Hsu, and Kuansan Wang. An overview of microsoft academic service (mas) and applications. *In Proceedings of the 24th International Conference on World Wide Web (WWW '15 Companion)*, 2005. URL https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/. (Cited on page 19.)

Digital Science & Research Solutions. Dimensions. https://www.dimensions.ai/, 2021. (Cited on pages 1 and 195.)

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. Mass: Masked sequence to sequence pre-training for language generation. In *International Conference on Machine Learning*, pages 5926–5936, 2019. (Cited on page 14.)

K. Sparck. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972. (Cited on pages 13 and 21.)

Karen Sparck Jones and Julia R Galliers. Evaluating natural language processing systems: an analysis and review. *Springer-Verlag*, 1996. (Cited on page 65.)

Flaminio Squazzoni, Giangiacomo Bravo, Francisco Grimaldo, Mike Garcia-Costa, Daniel Farjam, and Bahar Mehmani. Only second-class tickets for women in the covid-19 race. a study on manuscript submissions and reviews in 2329 elsevier journals. *Elsevier*, 2020. (Cited on page 1.)

M. Suneetha and F.S. Sameen. A feature terms based method for improving text summarization with supervised pos tagging. *International Journal of Computer Applications*, 47(23), 2012. (Cited on page 46.)

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf. (Cited on pages 29, 33 and 44.)

A. Tamura, K. Ishikawa, M. Saikou, and M. Tsuchida. Extractive summarization method for contact center dialogues based on call logs. *5th International Joint Conference on Natural Language Processing*, pages 8–13, 2011. (Cited on page 16.)

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *CoRR*, abs/2009.06732, 2020. (Cited on page 43.)

TensorFlow. c4, 2020. URL https://www.tensorflow.org/datasets/catalog/c4. (Cited on page 18.)

H. Thu. An optimization text summarization method based on naive bayes and topic word for single syllable language. *Applied mathematical sciences*, 8:99–115, 2014. (Cited on page 14.)

Juan-Manuel Torres-Moreno, Horacio Saggion, Iria da Cunha, Eric Sanjuan, and Patricia Velázquez-Morales. Summary evaluation with and without references. *Polibits*, 42:13–20, 12 2010. doi: 10.17562/PB-42-2. (Cited on pages 6, 51, 61, 62, 65, 66, 69 and 195.)

Raghuram Vadapalli, Litton J. Kurisinkel, Manish Gupta, and Vasudeva Varma. SSAS: semantic similarity for abstractive summarization. In Greg Kondrak and Taro Watanabe, editors, *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017, Volume 2: Short Papers*, pages 198–203. Asian Federation of Natural Language Processing, 2017. (Cited on pages 58, 63 and 75.)

N. Vanetik and M. Litvak. Query-based summarization using mdl principle. *Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, pages 22–31, 2017. (Cited on page 16.)

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.

(Cited on pages vii, 7, 14, 34, 36, 39, 40, 41, 42, 47, 49, 63, 91, 92, 93, 97, 98, 100, 103, 105, 109 and 112.)

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4566–4575. IEEE Computer Society, 2015. (Cited on page 58.)

Esaú Villatoro-Tello, Luis Villaseñor-Pineda, and Manuel Montes. Using word sequences for text summarization. In *9th International Conference, TSD 2006, Brno, Czech Republic*, pages 293–300, 09 2006. ISBN 978-3-540-39090-9. doi: 10.1007/11846406_37. (Cited on page 26.)

O. Vinyals, M. Fortunato, and Jaitly. N. Pointer networks. *Neural Information Processing Systems*, 2015. (Cited on page 48.)

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. Heterogeneous graph neural networks for extractive document summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6209–6219, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.553. URL `https://www.aclweb.org/anthology/2020.acl-main.553`. (Cited on page 72.)

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016. (Cited on page 41.)

Stratos Xenouleas, Prodromos Malakasiotis, Marianna Apidianaki, and Ion Androutsopoulos. SUM-QE: a bert-based summary quality estimation model. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6004–6010. Association for Computational Linguistics, 2019. (Cited on pages 61, 62 and 63.)

Qian Yang, Rebecca Passonneau, and Gerard De Melo. Peak: Pyramid evaluation via automated knowledge extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016. (Cited on page 58.)

Wonjin Yoon, Yoonsun Yeo, Minbyul Jeong, Bong-Jun Yi, and Jaewoo Kang. Learning by semantic similarity makes abstractive summarization better. *ArXiv*, abs/2002.07767, 2020. (Cited on page 72.)

Haoyu Zhang, Yeyun Gong, Yu Yan, Nan Duan, J. Xu, J. Wang, Ming Gong, and M. Zhou. Pretraining-based natural language generation for text summarization. In *CoNLL*, pages 789–797, 01 2019a. (Cited on page 72.)

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization, 2020a. (Cited on pages 4, 5, 7, 18, 41, 50, 51, 91, 93, 94, 96, 97, 98, 99, 100, 101, 102, 103, 104, 106, 109, 112 and 113.)

Rui Zhang and Joel Tetreault. This email could save your life: Introducing the task of email subject line generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 446–456, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1043. URL https://www.aclweb.org/anthology/P19-1043. (Cited on pages 19, 49, 50 and 91.)

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b. (Cited on pages 58, 61, 63, 70 and 75.)

Xingxing Zhang, Mirella Lapata, Furu Wei, and Ming Zhou. Neural latent extractive document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 779–784, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1088. URL https://www.aclweb.org/anthology/D18-1088. (Cited on page 72.)

Xingxing Zhang, Furu Wei, and Ming Zhou. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5059–5069, Florence, Italy, July 2019b. Association for Computational Linguistics. doi: 10.18653/v1/P19-1499. URL `https://www.aclweb.org/anthology/P19-1499`. (Cited on pages 17, 38, 47, 48 and 72.)

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 563–578. Association for Computational Linguistics, 2019. (Cited on pages 59, 63 and 70.)

Z. Zhenpeng. A hierarchical model for text autosummarization. `https://cs224d.stanford.edu/reports/zhenpeng.pdf`, 2016. (Cited on pages 14 and 34.)

Ming Zhong, Pengfei Liu, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Searching for effective neural extractive summarization: What works and what's next. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1049–1058, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1100. URL `https://www.aclweb.org/anthology/P19-1100`. (Cited on page 72.)

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6197–6208, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.552. URL `https://www.aclweb.org/anthology/2020.acl-main.552`. (Cited on page 72.)

Deyu Zhou, Linsen Guo, and Yulan He. Neural storyline extraction model for storyline generation from news articles. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1727–1736, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi:

10.18653/v1/N18-1156. URL https://www.aclweb.org/anthology/N18-1156. (Cited on page 72.)