UNIVERSITÉ PARIS XIII – SORBONNE PARIS NORD

École doctorale Sciences, Technologies, Santé Galilée

---

# Artificial Intelligence for the Configuration and Optimization of Digital Advertising Campaigns

Intelligence Artificielle pour la configuration et l'achat de campagnes publicitaires en ligne

---

THÈSE DE DOCTORAT
Présentée par


**Issam BENAMARA**
Laboratoire de Traitement et Transport de l'Information (L2TI)


pour l'obtention du grade de
DOCTEUR EN INFORMATIQUE

soutenue le 18-09-2024 devant le jury d'examen constitué de :

| | | |
|---|---|---|
| GUIGUE Vincent | AgroParisTech | Rapporteur |
| HEUTTE Laurent | Université de Rouen Normandie | Rapporteur |
| MOKRAOUI Anissa | Université Sorbonne Paris Nord | Examinatrice |
| VIENNET Emmanuel | Université Sorbonne Paris Nord | Directeur de thèse |
| VANNOD Bruno | The Programmatic Company | Invité |

# Acknowledgements

"Why don't you pursue a PhD?" A question that seemed so random when I heard it. My master's teacher at the time, Vincent Guigue, casually mentioned it during my master's defense. Little did I know that this suggestion would motivate me to actually pursue a PhD and ultimately lead me to where I am today. I want to thank Vincent for his guidance and support that made this thesis happen.

I owe a deep debt of gratitude to my supervisor, Emmanuel Viennet, for his wise and measured guidance throughout this journey. He struck the perfect balance between offering support and allowing me the independence to challenge myself and find my own solutions. His approach fostered both personal and professional growth, as he trusted me to navigate the complexities of my research while stepping in decisively when needed. Emmanuel's kindness, understanding, and patience created a stress-free environment that made this demanding process far more manageable, and for that, I am eternally grateful.

A special acknowledgment goes to Bruno Vannod, my boss, mentor, tutor, and father figure at The Programmatic Company. Bruno is the primary reason this PhD became a reality. His unwavering belief in me, coupled with his relentless push for me to reach greater heights, shaped me both as a scholar and a professional. Our deep, intricate discussions, often starting with work and business and evolving into philosophical explorations, challenged me to master my craft and view my work in a broader context. Bruno gave me purpose and pushed me toward excellence, and for that, I will always be deeply thankful.

I would like to extend my heartfelt thanks to L2TI, which provided me with an exceptional environment to carry out my research, particularly during the challenging times brought by the COVID-19 pandemic. A special thanks goes to Anissa Mokraoui, the director of L2TI and my tutor for this thesis. Her support and leadership ensured that I had all the resources and encouragement I needed to navigate these difficult circumstances and stay focused on my work.

I am deeply grateful to my friends and fellow PhD candidates, whose support, shared experiences, and valuable advice made each step of this journey more manageable. Their insights and reassurance during uncertain times were a source of comfort, helping me calm my fears and persevere. I want to express special thanks to Sara Bouguelia, who was always a step ahead. Her ability to take everything in stride while achieving success had a tremendous impact on my confidence and nerves. A heartfelt thanks also goes to Zack, my trusted brainstorming buddy, whose passionate discussions and genuine interest in every concept I explored made him an essential part of my intellectual journey—treating my challenges as if they were his own.

I would also like to thank my close friends, Adlane, Zinou, Salim, Hocine and Ismael, for being there when I needed to step away from my research, offering me moments of joy and simplicity that helped me recharge and return with fresh energy. Without failing to mention Wissal and Mohammed for their continuous support throughout this journey.

A very special thanks to Aghis, who listened to my evolving ideas week after week, even when they were half-formed or seemed nonsensical. Your patience and willingness to be my sounding board throughout this process was invaluable.

My deepest and utmost thanks go to my beautiful and dear wife, Sarah. Her unwavering support, despite managing her own projects, along with her endless care, love, and patience, has been my greatest source of strength throughout these years. Her sacrifices and understanding allowed me to pursue this demanding journey, and I truly could not have completed this PhD without her by my side. I love you my cutiepie <3.

I dedicate this work to my amazing parents, Nadia and Youssef, who have shaped me into the person I am today. Their education, undying support, and deep love have been the foundation of all my achievements. I also dedicate this to my sisters, Mouna and Insaf, who have always believed in me and supported me in every way possible. Without their encouragement, this journey would not have been the same. oh and, A HUGE KISS TO ANAS ! The most focused spongebob baby ever.

A huge thank you to Mama Lynda, Papa Mohammed, Lady Miriam, and King Adam for their love, support, and presence throughout this journey. AND ANOTHER HUGE KISS TO ELYA AND NAILA !

Last but not least, I want to extend my gratitude to all the organizations that contributed to the funding of this CIFRE thesis, particularly The Programmatic Company and the Association Nationale de la Recherche et de la Technologie (ANRT). The CIFRE framework provided me with the perfect opportunity to conduct research directly within an industrial environment, an invaluable experience that has shaped the outcome of this work.

# Abstract

In the evolving landscape of modern marketing, digital advertising has emerged as a pivotal component, enabling businesses to expand their reach to larger, more diverse audiences with unprecedented precision. Digital advertising platforms offer advantages such as controllable costs, accurate audience targeting, and measurable feedback. However, the escalating complexity of digital advertising ecosystems poses significant challenges in optimizing the performance of advertising campaigns. Traditional methodologies, heavily reliant on human expertise, are increasingly inadequate in addressing the multifaceted nature of these digital environments. Consequently, there is a growing dependence on algorithmic solutions and artificial intelligence (AI) to navigate this complexity and enhance campaign outcomes.

Current works predominantly focus on post-launch campaign optimization, such as key performance indicator (KPI) and cost forecasting to optimize real-time bidding (RTB) agents which aim to refine budget allocation and maximize the effectiveness of ongoing campaigns. Despite the achieved progress, the effectiveness of these methods depends on the accurate configuration of advertising campaigns, specifically in terms of targeting the appropriate audiences with the correct parameters. A task that we call Advertising Strategy Design.

Unfortunately, due to the inherent scale and complexity of the task, there is a noticeable gap in pre-launch campaign optimization process. Advertising strategy design still relies heavily on human expertise, which often leads to sub-optimal targeting and decision-making. This affects the overall campaign performance, underscoring a potential area of improvement.

This thesis aims at leveraging artificial intelligence methods for the configuration and optimization of digital advertising campaigns. For this purpose, we integrate deep learning approaches in the initial phases of campaign planning in the task of advertising strategy design. In this thesis, we first contribute a novel framework and generative neural network model which leverages the attention mechanism through transformers to contextually generate optimal advertising strategies while avoiding combinatorial explosion. We evaluate our results on a public dataset iPinYou as well as the company's private dataset by measuring the closeness of the generated strategies to the datasets (using Cosine and Hamming distances) as well as their estimated KPI performance. In the absence of directly comparable methods, we benchmarked our results against prominent methods from other fields, adapted for this specific task. We further refined our approach by enhancing the generative diversity, improving robustness against mode collapse—a condition where the model tends toward generating a limited range of outputs—and introducing an inference-time exploration mode employing vector quantization techniques and learned metrics. An improved evaluation protocol for our framework was also developed. We finally propose a novel token-driven methodology for flexible generative control in transformer-based models. This method includes a suggestive input mechanism that allows the model to take user preferences into account while maintaining the freedom to deviate from them if they do not lead to optimal outcomes, treating these inputs as guiding suggestions rather than strict rules. Extensive experiments were conducted to assess the effectiveness of our approach, which yielded outstanding results and confirmed its applicability across various domains utilizing transformer models.

# Résumé


Avec l'évolution du marketing moderne, la publicité numérique est apparue comme un composant central, permettant aux entreprises d'étendre leur portée à des audiences plus larges et plus diversifiées avec une précision sans précédent. Les plateformes de publicité numérique offrent des avantages tels que des coûts contrôlables, un ciblage d'audience précis et la mesurabilité des performances. Cependant, la complexité croissante des écosystèmes de publicité numérique pose des défis significatifs dans l'optimisation des campagnes publicitaires. Les méthodologies traditionnelles, fortement dépendantes de l'expertise humaine, sont de plus en plus inadéquates pour aborder la nature complexe de ces environnements numériques. Par conséquent, il y a une dépendance croissante aux solutions algorithmiques et à l'intelligence artificielle (IA) pour naviguer cette complexité et améliorer les résultats des campagnes.

Les travaux actuels se concentrent principalement sur l'optimisation des campagnes après leur lancement, telles que la prévision des indicateurs clés de performance (KPI) et des coûts pour optimiser les agents de bidding en temps réel (RTB) qui visent à affiner l'allocation budgetaire et maximiser l'efficacité des campagnes en cours. Malgré les progrès réalisés, l'efficacité de ces méthodes dépend de la bonne configuration préalable des campagnes publicitaires, spécifiquement en termes de ciblage d'audiences et un bon paramètrage. Une tâche que nous appelons Conception de Stratégie Publicitaire.

Malheureusement, en raison de l'échelle et de la complexité inhérentes à la tâche, il y a un manque considérable dans l'optimisation des campagnes avant leur lancement. La conception de stratégie publicitaire repose encore fortement sur de l'expertise humaine, ce qui conduit souvent à un ciblage sous-optimal, et dégrade la prise de décision ainsi que la performance globale de la campagne, soulignant un domaine potentiel d'amélioration.

Cette thèse vise à exploiter les méthodes d'intelligence artificielle pour la configuration et l'optimisation des campagnes publicitaires numériques. À cette fin, nous intégrons des approches d'apprentissage profond dans les phases initiales de planification de la campagne dans la tâche de conception de stratégie publicitaire. Dans cette thèse, nous contribuons d'abord un système novateur et un modèle de réseau de neurones génératif qui exploite le mécanisme d'attention des transformers pour contextuellement générer des stratégies publicitaires optimales tout en évitant l'explosion combinatoire. Nous évaluons nos résultats sur un ensemble de données public iPinYou ainsi que sur les données de l'entreprise en mesurant la proximité des stratégies générées avec les ensembles de données (Distance Cosinus et Hamming) ainsi que leur performance KPI estimée. En l'absence de méthodes directement comparables, nous avons comparé nos résultats à des méthodes principales d'autres domaines, adaptées à cette tâche spécifique. Nous affinons ensuite notre contribution, en améliorant la diversité générative, en améliorant la robustesse contre le mode-collapse — une condition où le modèle tend à générer une gamme limitée de sorties — et en introduisant un mode exploratoir au moment de l'inférence via des techniques de quantification vectorielle et l'apprentissage de métriques. Nous proposons également un protocole d'évaluation amélioré pour notre système. Nous proposons finalement une méthodologie novatrice axée sur des tokens de signalisation pour le contrôle génératif flexible dans les modèles basés sur les transformers qui prends en entrée des signaux suggestifs, ce qui permet à notre modèle de considérer les préférences utilisateur tout en conservant l'autonomie de s'en écarter si elles ne produisent pas de résultats optimaux, les intégrant dans le processus génératif comme des paramètres suggestifs plutôt que des directives strictes. Des expériences étendues ont été menées pour évaluer l'efficacité de notre approche, qui a produit des résultats exceptionnels et confirmé son applicabilité dans divers domaines utilisant des modèles de transformateurs.

# Contents

# List of Figures

# List of Tables

# Publications

## Conferences

- **Issam Benamara** and Emmanuel Viennet, "Contextual Advertising Strategy Generation via Attention and Interaction Guidance" in 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA), 2023.

## Workshops

- **Issam Benamara** and Emmanuel Viennet, "Contextual Advertising Strategy Generation via Attention and Interaction Guidance" in Atelier sur les mécanismes d'attention en apprentissage automatique de la conférence EGC, 2023.

## Papers in Preparation

- **Issam Benamara** and Emmanuel Viennet, "Leveraging Quantization for Controllable Diversity and Exploration in Advertising Strategy Generation".
- **Issam Benamara** and Emmanuel Viennet, "Power of Suggestion: Strategic Feature Manipulation in Transformer-Based Models " in AdKDD 2024 Workshop in conjunction with The 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024. **(submitted)**

# Chapter 1

# Introduction

## 1.1 Context and Motivation

This PhD thesis emerges from a partnership between the L2TI laboratory at Université Sorbonne Paris Nord (USPN) and The Programmatic Company[1], an innovative firm offering a comprehensive suite of tools designed to centralize and automate the management of digital media campaigns.

For more than $209.7B in revenue in 2022 [IAB, 2023], the continuous growth of the digital advertising industry in terms of market size and audience targeting opportunities makes it an essential component of modern marketing. This important growth is due to the increasing usage of digital devices and the broadened access to the Internet, leading to more opportunities for businesses to reach their target audience with greater precision compared to traditional advertising methods, such as billboards. Digital advertising platforms (Diffusion platforms) enable advertisers to meticulously craft campaigns tailored to specific demographic segments based on a variety of criteria, from geographical location to personal interests. On top of accurate audience targeting, these platforms allow for controllable costs and measurable feedback.

The escalating complexity of digital advertising ecosystems poses significant challenges in optimizing the performance of advertising campaigns. Therefore, reliance on human expertise is becoming increasingly inadequate in addressing the multifaceted nature of these digital environments. Consequently, there is a growing dependence on algorithmic solutions and artificial intelligence to navigate this complexity and improve campaign performance.

Digital advertising works by displaying ads on websites, social media platforms, and other digital channels. When a business wishes to advertise their product, they have to set up an advertising campaign. They define the goal of the campaign (e.g. increase sales, brand awareness, etc.). They identify the audiences to target by specifying multiple features

[1] `https://www.theprogrammaticcompany.com/`

(e.g. country, age, interests, etc.). Audiences are chosen on the basis of how likely they are to be interested in the product. Finally, they choose one or many diffusion platforms (e.g. Instagram, Google Search, websites, etc.) that will display the ads. Diffusion platforms are where advertisers choose more precisely who, when, and how to target by specifying diffusion dates for the ad campaign or more technical features, such as real-time bidding parameters. In this thesis, we use the term *"advertising strategy"* to refer to a combination of all feature instances (e.g. France, Women, 18-25 years old, Interested in gaming, Click Optimization Goal, Banner Ad, Placement ID59, etc.). Google refers to this combination as a *"line item"* [Google, 2024a] but we generalize the concept and use the term advertising strategy regardless of the terminology used on other platforms. An advertising campaign is composed of multiple such advertising strategies, all designed to collectively work toward accomplishing a singular overarching campaign objective.

Real-Time Bidding (RTB) is a programmatic advertising technique where advertisers bid in real-time for ad impressions as they become available [Amazon Ads, 2024]. Winning a bid in this auction allows an advertiser's ad to be displayed on the publisher's site. RTB enhances efficiency and enables advertisers to target the most relevant inventory. In the context of RTB, a Demand-Side Platform (DSP) enables advertisers to automate the purchasing of ad inventory, manage multiple ad exchange and data exchange accounts through one interface, and optimize the performance of their ads. A Data Management Platform (DMP) collects and organizes data from various sources, providing insights and segments to advertisers, which can be used to refine targeting strategies through the DSP. The Supply-Side Platform (SSP) is used by publishers to manage their ad inventory and maximize revenue by automating the sale to the highest bidder in real-time auctions.

Current works primarily focus on optimization after a campaign has been launched. In Real-Time Bidding (RTB), automated tools employed by advertisers utilize forecasting models for Key Performance Indicators (KPIs), such as Click-Through Rate (CTR, the percentage of users who click on an advertisement after viewing it) or Cost-per-Mile (CPM, the cost for a thousand impressions or views), to optimize their bidding agents.

These agents strategically bid in each auction to better allocate the budget and enhance the performance and effectiveness of ongoing campaigns. Despite these advancements, the success of these post-launch methods heavily relies on an optimized configuration of advertising campaigns before their launch, especially in terms of accurately targeting the right audiences with the appropriate parameters. We call this crucial task *Advertising Strategy Design*.

Despite the importance of Advertising Strategy Design, this process still heavily relies on human experts, utilizing prior experience and intuition to navigate the selection of audience targeting features and campaign parameters. This approach, while grounded in practical knowledge, is fraught with inefficiencies, requiring extensive time and resources and often resulting in less than optimal campaign outcomes [Blakeman, 2015]. Unfortunately, due to the inherent scale and complexity of the task, there is a noticeable gap in pre-launch campaign optimization research as the vast majority of current research focuses on optimization occuring after campaigns are already configured and launched (e.g. RTB bidding agent optimization). Developing an effective advertising campaign is complex due to the multitude of variables involved, as detailed in Section 1.2. A key challenge is the combinatorial nature of advertising strategies, where the success of a strategy depends entirely on the entire combination of its features. Consequently, methods that construct advertising strategies incrementally, one item at a time, are inherently suboptimal. This is because there is no prior knowledge about the performance of a partial combination of items, making it difficult to predict the effectiveness of these incomplete strategies.

Contextual advertising strategy generation can be described as follows: given a context (e.g. advertiser industry, diffusion platform, etc.) and a campaign goal KPI (e.g. Cost Per Click), generate a set of advertising strategies that perform the best in terms of the goal KPI. For instance, if the goal is to reduce the Cost Per Click (CPC), the aim is to generate strategies that yield more clicks for the same budget, thereby potentially increasing sales without increasing spending.

In pursuing this research, we identified key constraints essential for successfully navigating the complexities of digital advertising strategy design (see Chapter 4). Central to these challenges was managing the extensive feature space of advertising strategies. This required our methods to avoid exhaustive and combinatorial exploration without resorting to filtering methods, which could overlook potentially valuable strategies without prior knowledge of their impact. Due to the atomic nature of each strategy—where its effectiveness is directly linked to the specific combination of its features—the sequential generation of strategy elements proved suboptimal. This underscored the necessity for order-agnostic processing and precise modeling of feature interactions. Ensuring data fidelity was essential, as the generated strategies must be trustworthy and closely aligned with established data points. This aspect of data fidelity should be adjustable to meet the diverse needs of various clients and to appropriately balance exploration and exploitation as required. Additionally, as detailed in Chapter 6, the integration of AI-assisted methods introduced further constraints. The approaches needed to regard user preferences as flexible suggestions that could influence the generation of strategies, rather than as rigid mandates. This would allow for adaptable conditioning, enabling the injection of guidance and preferences into the strategy generation process while maintaining the autonomy to deviate from these preferences if they do not lead to optimal outcomes.

This thesis aims at leveraging artificial intelligence methods for the configuration and optimization of digital advertising campaigns. For this purpose, we integrate deep learning approaches in the initial phases of campaign planning in the task of advertising strategy design. In this thesis, we first contribute a novel framework and generative neural network model which leverages the attention mechanism through transformers to contextually generate optimal advertising strategies while avoiding combinatorial explosion (see Section 1.3.1). We evaluate our results on a public dataset iPinYou [Liao et al., 2014] as well as the company's private dataset by measuring the closeness of the generated strategies to the datasets (using Cosine and Hamming distances) as well as their estimated KPI performance. In the absence of directly comparable methods, we benchmarked our results against prominent methods from other fields, adapted for this specific task. We further refined our approach by enhancing the generative diversity, improving robustness against mode collapse—a condition linked to variational components where the model tends toward generating a limited range of outputs—and introducing an inference-time exploration mode employing vector quantization techniques and learned metrics (see Section 1.3.2). An improved evaluation protocol for our framework was also developed. We finally propose a novel token-driven methodology for flexible generative control in transformer-based models (see Section 1.3.3). This method includes a suggestive input mechanism that allows the model to take user preferences into account while maintaining the freedom to deviate from them if they do not lead to optimal outcomes, treating these inputs as guiding suggestions rather than strict rules. Extensive experiments were conducted to assess the effectiveness of our approach, which yielded outstanding results and confirmed its applicability across various domains utilizing transformer models.

## 1.2 Scientific and Industrial Challenges

In this section, we will discuss critical considerations related to the task of digital advertising strategy design, addressing the challenges it presents across both scientific research and industrial practice.

### 1.2.1 Scientific Challenges

Advertising Strategy Design is fundamentally a task of generation and recommendation. Beyond the established challenges inherent to generative and recommendation systems, this specific task introduces additional constraints that are primarily derived from its unique characteristics. We highlight the principal challenges that significantly influence the efficacy and outcomes of any approach, and more specifically our approaches.

Firstly, **Data-Related Challenges** have a significant impact due to the lack of standardized data formats across various platforms and sources. This inconsistency often converts simple tabular data into more complex tree structures by aggregating strategy features into lists, which complicates data handling and necessitates sophisticated methods tailored for non-tabular data. In this thesis, we limit our focus to tabular data to simplify the complex task of advertising strategy generation.

Advertising datasets typically contain a high proportion of categorical variables, such as geographical locations, demographic groups, interests, and ad placements. These datasets pose challenges for modeling, particularly in output generation, because many conventional methods that model distributions and sample indices may not effectively capture the intricate relationships and hierarchies within these variables [Sar Shalom et al., 2016, Deng et al., 2021, Ban et al., 2021, Katz et al., 2022]. Furthermore, the high dimensionality of feature spaces in these datasets often leads to combinatorial explosion when attempting exhaustive search. For instance, selecting just 8 out of the 24 features in the iPinYou Dataset can generate over 678 million possible combinations. This makes it difficult to efficiently explore and optimize strategy space—a central objective of our research.

Additionally, the datasets primarily consist of historical data derived from previously optimized advertising campaigns. This results in a concentration of data around a limited set of historically successful strategies, creating data selection bias and leaving many feature attributes under-explored. This problem is intensified by missing critical features in the datasets. For instance, two strategies might appear identical in the dataset, sharing the same attributes yet displaying significantly different costs (or any other KPI). This discrepancy can arise from differences in how the strategies were setup on advertising platforms but not shown in the dataset, such as the "Skippable Ad" feature, which can significantly influence ad cost but may be omitted by data providers, leading to noisy datasets where seemingly identical strategies perform differently in practice.

Secondly, **Recommendation Task Related Challenges** are also significant. Unlike traditional recommender systems that typically rank individual items, our task involves recommending several interdependent items simultaneously—a process known as itemset recommendation. The entire advertising strategy must be generated at once, without any hierarchical or sequential order among features. It's challenging to predict the performance of a strategy accurately from just a subset of its features, as altering even a single feature (such as making an ad skippable) can drastically affect the strategy's success.

Moreover, since the objective is to generate a variety of strategies within a single context, employing variational methods to enhance output diversity is essential. However, these methods introduce complexities like mode collapse [Takida et al., 2022,Liu et al., 2023a], where the model fails to produce a diverse range of outputs and instead converges on generating a limited set or even a single output repeatedly, leading to insufficient diversity. Related to the challenge of output diversity is the need to balance exploration and exploitation. It is crucial to find a balance between leveraging known successful strategies and venturing into new, potentially beneficial ones without significant risk, which involves managing uncertainty.

The complexity of this task demands a sophisticated methodological approach. Integrating multiple models into a cohesive framework adds further complexity to the training process, requiring harmonization of different data representations, scales, and learning behaviors. Additionally, akin to challenges faced by traditional recommender systems, it is vital to establish reliable evaluation protocols to gauge the quality and effectiveness of such frameworks. This involves assessing the quality and reliability of the generated strategies, quantifying the diversity among recommended strategies, and evaluating their trustworthiness.

### 1.2.2 Industrial Challenges

Several research challenges in digital advertising strategy design closely mirror the obstacles faced in the industry. This discussion highlights the primary issues from both business and technical perspectives.

Many of these challenges stem from **Data-Related Issues**. Acquiring high-quality digital advertising data typically involves conducting ad campaigns or buying data from external sources, which can be prohibitively expensive. This cost is compounded by the expenses of data cleaning and integration. Additionally, managing and processing the large volumes of data typical in digital advertising poses significant technical challenges, demanding considerable resources for data analysis, storage, and retrieval. Training complex deep learning models on such vast data sets requires extensive computational power, making the process both resource-intensive and costly.

Compliance with data privacy regulations and ethical standards is also critical in digital advertising. Strict regulations, such as the General Data Protection Regulation (GDPR), underscore the necessity of managing personal data responsibly. These regulations often result in the exclusion of potentially predictive features from datasets, complicating data analysis and modeling. Furthermore, advertising platforms might withhold certain data to preserve competitive advantages or control over their ecosystems, adding another layer of complexity. Efforts to recover or approximate these missing features can be resource-intensive, demanding complex approaches to infer or simulate withheld information.

Once models are developed, deploying them into production presents its own set of **Deployment and Operational Challenges**. When advertising strategies are recommended or generated, the process must be fast enough comfortable use to avoid service-quality impairments due to delays or latency [OpenAI, 2024]. Particularly in automated environments, models must strike a balance between complexity and the necessity for rapid inference. Post-deployment, it is essential to monitor the quality of model outputs. While offline evaluations of recommendations aim to statistically validate model predictions, deploying these recommendations online introduces uncertainties due to the dynamic nature of real-world environments, potentially affecting outcomes. Testing and assessing advertising strategies in real-time scenarios involve financial costs and risks, including possible negative impacts on brand reputation if not carefully managed.

## 1.3 Contributions

In this section, we outline the contributions of our work to the field of digital advertising strategy design, highlighting the role of machine learning in facilitating these advancements.

Our primary objective is to enhance advertising campaigns right from the planning stage, where marketers begin to convert their business and marketing briefs into an advertising media plan. We achieve this optimization by recommending a set of diverse and high-value advertising strategies that will form the campaign. Here, "valuation" is determined by the campaign goals; for example, if the objective is to boost sales, an effective measure would be to optimize the overall number of clicks, thus advertising strategies with a high click-through rate (CTR) would be considered high-value or high-scoring strategies.

Additionally, we employ an AI-assisted approach to enable customization of the generated strategies based on user preferences. This customization is achieved by providing options that balance exploration and exploitation, allowing users to choose between a reliable, trustworthy generation path and a more innovative, experimental approach. Furthermore, we incorporate a flexible conditioning mechanism that enables users to suggest desired feature attributes in the strategies, provided these attributes contribute to creating high-value outcomes.

Our journey towards realizing this ambitious goal unfolds across three major milestones:

- Initially, as introduced in Section 1.3.1 and later detailed in Chapter 4, we tackle the main challenge of combinatorial explosion by introducing a transformer-based model within a novel framework designed for guided non-autoregressive generation.
- Subsequently, as introduced in Section 1.3.2 and later detailed in Chapter 5, we aim to enhance output diversity and robustness in our methodology. We propose an exploratory mode activated at inference and dedicate efforts to refining the

evaluation protocols to better assess the quality and effectiveness of our methods.

- Finally, as introduced in Section 1.3.3 and later detailed in Chapter 6, we concentrate on implementing a suggestive control mechanism that seamlessly incorporates individual preferences into the generative process, acting as flexible guidelines rather than strict directives.

The upcoming sections outline the three main contributions of this research. Each section introduces the core ideas and methodologies behind these contributions, setting the stage for a deeper examination in subsequent chapters.

### 1.3.1 Contextual Advertising Strategy Generation via Attention and Interaction Guidance

Contextual advertising strategy generation can be described as follows: given context features (e.g. advertiser industry, diffusion platform, etc.) and a campaign goal KPI (e.g. Cost Per Click) that serves as a performance score, generate a set of advertising strategies that perform the best in terms of the goal KPI. A single advertising strategy is composed of strategy features (e.g. Country, Device, Gender...).

One of the primary challenges of this task is the high dimensionality of strategy features, leading to a combinatorial explosion that renders the exhaustive exploration of all possible strategy combinations unfeasible. Traditional recommender systems falter in the face of this combinatorial complexity, and many generative approaches either rely on suboptimal auto-regressive processes, fail to capture complex feature interactions effectively, or do not consider overall strategy performance during generation.

Addressing these challenges, we introduce a novel method and framework specifically designed for the contextual generation of advertising strategies. Similarly to traditional recommender systems, we start by training an adapted state-of-the-art Click-Through Rate prediction model (called the *Estimator*) which processes a concatenated input of context and strategy features to predict a utility score reflecting a Key Performance Indicator (KPI). This model is responsible for capturing complex feature interactions.

We then design a transformer encoder decoder (called the *Generator*) which takes as input the context features alongside the whole vocabulary of strategy features and outputs strategy features as a single combination in one shot. This combination is then scored through the frozen *Estimator*. This serves as a guiding signal for the attention mechanism in the *Generator* to learn to focus on the most interesting features for each context and how to combine them. We use a smooth contrastive learning method to train the *Generator* via a novel loss function. This loss function leverages the *Estimator*'s guidance as well as other parameters to tune the *Generator*'s data fidelity, to either generate robust and likely strategies close to the data, or generate new and promising strategies by being more exploratory. A VAE-like variational component [Fang et al., 2021] and consistent dropout use were employed to enhance output diversity. This approach is responsible for dealing with the combinatorial explosion problem, considering strategy performance during generation, one shot strategy generation and provides an exploration/exploitation balancing choice.

We primarely evaluate our results on a public dataset iPinYou [Liao et al., 2014] by measuring the closeness of the generated strategies to the datasets (Cosine & Hamming) as well as their estimated KPI performance. We also trained our models on the company's private dataset and deployed this method into production.

This contribution demonstrated superior results, outperforming other approaches while adhering to the task's constraints and providing adjustable exploration/exploitation options essential for meeting diverse client needs.

### 1.3.2 Leveraging Quantization for Controllable Diversity and Exploration in Advertising Strategy Generation

It is essential for marketers to have control over the level of exploration in advertising strategy generation to manage investment risks effectively while uncovering new and effective advertising strategies. This requirement underpins the development of an AI-assisted approach to advertising strategy generation. This contribution focuses on enhancing the strategy diversity and robustness of our previous methods and on providing a customizable balance between exploration and exploitation, ensuring it accommodates the diverse needs of various clients.

Controllable fidelity pertains to the model's capacity to regulate the degree of alignment between generated strategies and historical data. This feature allows users to dictate whether the model prioritizes strategies that closely mirror past successes (thereby deemed more trustworthy) or encourages the exploration of novel promising strategies. Such flexibility is crucial for balancing the exploitation of known effective strategies against the exploration of innovative ones that could uncover new opportunities or efficiencies.

In this contribution, we propose replacing the VAE-like variational component [Fang et al., 2021] from our previous work with vector quantization to address the issue of mode collapse. This change not only increases robustness by mitigating mode collapse but also enhances diversity of the generated strategies. Additionally, the introduction of quantized tokens adds a controllable fidelity mechanism by providing more or less likely quantized tokens as in [Kolesnikov et al., 2022], facilitating a shift towards more exploratory strategy generation during inference without the need for retraining the base model.

One notable shortcoming of our earlier method was its inability to sensitively respond to minor changes in strategy features, which could significantly diminish model performance. To overcome this, we have introduced a new neural network component, named the *Aligner*, which evaluates the similarity or alignment between two strategies by assessing how their differences impact strategy-valuation. This component serves a similar function to the *Estimator* model from our earlier efforts, providing a crucial guiding signal during the *Generator* model's training to ensure small feature adjustments are effectively captured and incorporated into strategy generation.

Moreover, we have enhanced our evaluation protocol by integrating new metrics and processes that allow for a more precise assessment of our methods' effectiveness and quality. As a result, we observed that our approach has successfully mitigated mode collapse and achieved a notable increase in generative diversity. The exploratory mode during inference time has shown to be effective, delivering excellent results without compromising the overall performance, which is essential for clients interested in exploring new strategies without the necessity of a dedicated model trained in exploratory mode. The enhanced evaluation protocol, with clearer performance metrics, has confirmed the effectiveness of our methods and provided deeper insights to further refine the training process.

### 1.3.3 Strategic Feature Manipulation in Transformer-Based Models: A Novel Token-Driven Methodology

When marketers utilize a recommendation system or a generative model for advertising strategy design, they usually need the flexibility to input their preferences in two distinct ways. Firstly, they may want to enforce certain preferences strictly, ensuring that these are always incorporated into the generated strategies. Secondly, they may prefer to provide preferences more flexibly, merely as suggestions that guide the generative model. This allows the model the autonomy to decide whether adhering to these suggestions achieves optimal outcomes or if deviating from them might result in more effective solutions. This mechanism enables marketers to fine-tune the strategy generation process to balance creativity and precision effectively.

In our earlier work, we addressed the incorporation of strict preferences through contextual generation. In this contribution, we shift our focus to accommodating flexible preferences. In the context of advertising strategy generation, flexible preferences occur when users specify certain feature attributes they want included in the strategies or, conversely, which attributes should be avoided. This approach allows users to guide the generation process subtly, suggesting desired or undesired features without mandating their inclusion or exclusion.

The challenge involves striking a balance between conforming to user preferences and leveraging the model's acquired knowledge. Thus, the model can either generate an output that aligns with user preferences or independently diverge from these preferences when it deems them suboptimal, based on its learning. In essence, this approach is similar to seeking advice from a friend who, when presented with specific preferences, either offers guidance aligning with those preferences or, if unable to accommodate them, suggests an alternative they consider beneficial, implying the initial preferences may not be practical.

In this contribution, extending the foundation laid by our preceding efforts, we introduce an innovative token-driven approach designed to seamlessly incorporate suggestive preferences into the generative model. This methodology integrates specialized token embeddings alongside the embeddings of each strategy feature, thereby instructing the model on whether to prioritize, disregard, or remain neutral towards generating particular strategy feature attributes. To effectively train the model, we employ a scheduled token masking strategy, which allows the model to function in its standard mode under neutral token conditions or to actively seek to include or exclude specific attributes based on the presence of suggestive tokens. We also adjust the loss function of the Generator model to avoid some new extreme cases of mode collapse that occurs when attributes are negatively suggested (to be avoided).

This approach not only enhances the model's flexibility in generating strategies but also enables a more tailored generation process that aligns with user-induced preferences while maintaining the model's autonomy to optimize outcomes.

Extensive experiments were conducted to evaluate this approach, which produced outstanding results, demonstrating its viability across various domains utilizing transformer models. The model demonstrated strong adherence to user suggestions without compromising performance, which is crucial in a production environment for meeting client needs.

## 1.4 Thesis structure

Structured into seven chapters, this thesis methodically introduces and delineates our research's contributions to digital advertising strategy design. The organization and progression of these chapters are thoughtfully aligned with the research approach illustrated in Figure 1.1, ensuring a coherent presentation of our work. Chapter 2 establishes the academic context for our contributions by reviewing existing literature in the field of machine learning applied to digital advertising. Chapter 3 presents the data and technical challenges. The next three chapters (Chapter 4, 5, and 6) present the detailed methodologies and results for each contribution. Specifically, Chapter 4 proposes and presents the core methodology underlying our approach to contextual advertising strategy generation. Chapter 5 proposes and presents improvements over the previous chapter works towards a more controllable, diverse and reliable generative process. Chapter 6 proposes and presents a novel token-driven method for inducing suggestive preferences within the generative process. Finally, Chapter 7 concludes by providing highlights of key findings and discusses future research directions.

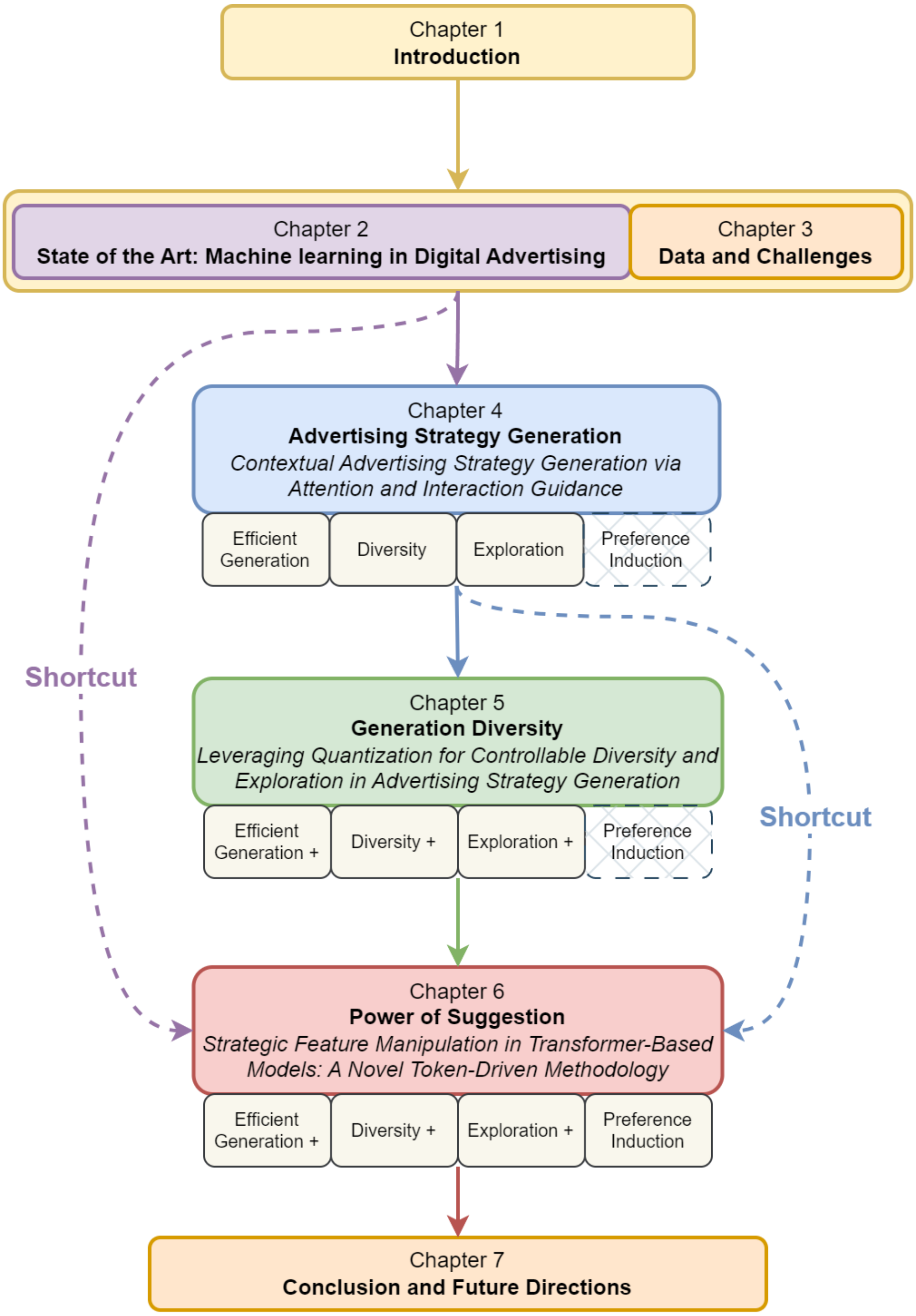



Figure 1.1: Research Approach.

# Chapter 2

# State of the Art: Machine Learning in Digital Advertising

## Contents

## 2.1 Introduction

The growing complexity of digital advertising systems has gained considerable academic and industrial attention. In this chapter, we explore the state of the art of machine learning applied in the domain of digital advertising. We delve into three foundational areas in this field: Click-Through Rate Forecasting (CTR, the percentage of users who click on an advertisement after viewing it), Real-Time Bidding (RTB) Strategies, and Advertising Strategy and Bundle Recommendation. The relationship between them is shown in Figure 2.1.

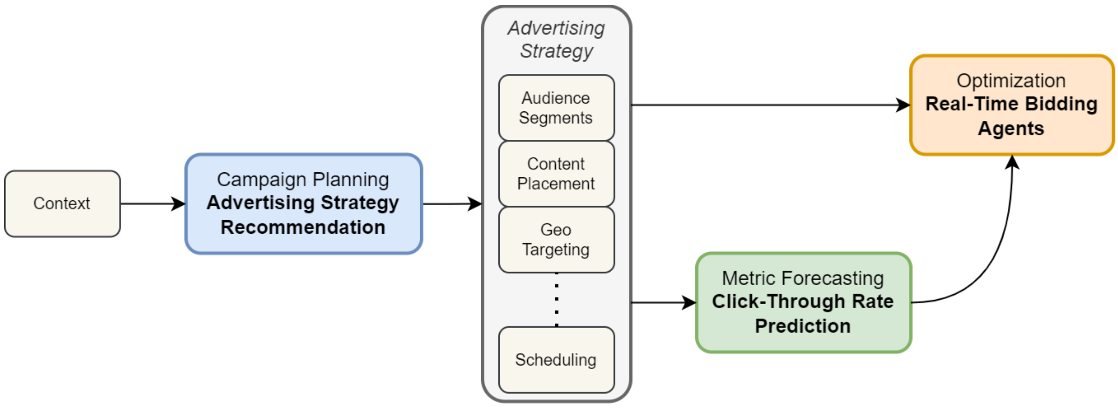



Figure 2.1: The relationships between the three main Digital Advertising tasks discussed in this thesis.

CTR prediction models form the backbone of performance metric forecasting, allowing the advertisers to gauge the potential value of ad placements. In section 2.2.1, we showcase the evolution of these models from basic logistic regression to complex deep learning architectures that capture complex feature interactions to improve performance. Real-Time Bidding Strategies make use of such performance metric forecasting models to learn how to act accordingly to each ad opportunity. In section 2.2.2, we explore the wide variety of sophisticated bidding agents along their shift from static to dynamic strategies.

While CTR prediction and RTB strategies are well documented, the task of Advertising

Strategy Recommendation remains heavily underexplored. Unlike traditional recommendation systems that suggest individual items and then rank them to recommend a list of independent items, this task involves recommending a set of items that are can only valued when viewed as a complete set (bundle). For example, the performance of an advertising strategy can only be assessed when all its attributes are specified; subsets of items cannot be valued independently. This integral evaluation adds a layer of complexity to the task. In section 2.2.3, we briefly review the main methods of recommender systems and explore the state of the art of of bundle recommendation as significant parallels can be drawn to this task.

The recent rise and success of Large Language Models (LLMs) [OpenAI, 2023, Touvron et al., 2023, Abdullah et al., 2022, Peebles and Xie, 2023, Rombach et al., 2022a] and image generation models [Rombach et al., 2022a, Peebles and Xie, 2023, Betker et al., 2023, Ramesh et al., 2021, Dhariwal and Nichol, 2021, Chang et al., 2022, Esser et al., 2021, Saharia et al., 2022] has marked a significant milestone in artificial intelligence. While the direct application of these technologies in digital advertising is not the core focus of this thesis, the advanced machine learning techniques proposed in these models have proven to be incredibly powerful and innovative. In section 2.2.4, we highlight the main concepts and techniques that were pivotal to this thesis.

In section 2.3, we summarize the strengths and limitations of the methods we explored in the state of the art sections, and we position our contributions compared to the existing approaches.

## 2.2 State of the Art

### 2.2.1 Click-Through Rate Prediction

One of the most important components in recommender systems is performance metric forecasting. Most notably used to estimate the value of generated candidates and ranking tasks [Covington et al., 2016a]. In online advertising, Click-through Rate (CTR) Prediction is a crucial task [Yang and Zhai, 2022, Chen et al., 2016]. It aims to predict the probability of a user clicking on a recommended item or an advertisement on a web page. Accurate CTR prediction often leads to better advertising performance as the advertising budget is optimized to be spent on the most promising adverts. Thus bringing significant revenue gains and also improved user satisfaction [Cheng et al., 2016, Wang et al., 2021b].

In recommender systems and online advertising, features such as user demographic details, user interests and behavior patterns, ad placement, and contextual data about the interaction environment (such as time and location) are pivotal. Most methods of Click-through Rate (CTR) prediction typically involves three key components: feature embedding, feature interaction, and prediction. Feature Embedding involves converting the raw features into a more manageable form, typically dense vectors, which are easier for machine learning models to process. This transformation is crucial because it helps to reduce the dimensionality of data, especially for categorical features, and ensures that the essential information is encapsulated in a form that optimally feeds into further processing stages. Feature Interaction is critical in modeling how different features influence each other and, consequently, the prediction outcomes. Feature interaction explores the relationships between pairs or groups of features, determining how these combined features can affect user behavior like clicking on an ad. This component is extensively researched due to its significant impact on improving the accuracy of predictive models [Yang and Zhai, 2022, Chen et al., 2016].

Earlier works such as Logistic Regression (LR) [Richardson et al., 2007], and Factorization Machine (FM) [Rendle, 2010] based methods [Blondel et al., 2016, Juan et al., 2016, Guo et al., 2017, He and Chua, 2017] proposed to learn both low and high order feature interactions. FMs model all interactions between variables using factorized parameters, hence the name. Unlike standard linear models that sum the effect of individual variables independently, FMs can estimate interactions between any two features in a linear time complexity. This is achieved by representing each variable with a latent vector and modeling interactions as the dot product of these latent vectors.

FM-based methods, as highlighted in sources such as [Blondel et al., 2016, Juan et al., 2016, Guo et al., 2017, He and Chua, 2017], utilize a dual-component approach to model feature interactions: the FM part and a feed-forward neural network, often referred to as the deep part. Typically, the FM component is responsible for learning second-order feature interactions, which is particularly effective for sparse data modeling, while the deep part handles higher-order interactions. Subsequent advancements in FM-based methods, as seen in works like [Xiao et al., 2017, Pan et al., 2018], have enhanced performance by assigning weights to feature interactions based on their significance.

Despite these advancements, FM-based methods generally focus on capturing only low-order interactions within the FM component due to the exponential complexity involved in higher-order interactions, leaving more complex interactions to the deep component. More recent developments, such as those discussed in [Yu et al., 2020], introduce innovative ways to represent higher-order FMs using various mathematical approximations to lower complexity. However, these approaches come with limitations: they often struggle to effectively model higher-order interactions without significantly increasing model complexity and potential scalability issues. Additionally, FM methods typically do not accommodate temporal dynamics in data and may make it difficult to interpret the contribution of individual features and their interactions to the final prediction, making it challenging to extract actionable insights and understand model behavior.

Moreover, other deep learning approaches have shown superior performance in capturing complex non-linear interactions and high-order feature combinations. Such methods focused on designing effective feature interaction architectures. The works of [Wang et al., 2017b, Lian et al., 2018], proposed to explicitly model pairwise and high order feature interactions via a sub-network (interaction network) parameterized to capture up to the k-th order interactions. Such sub-networks advantages are the explicitly applied automatic feature crossing and their easy integration with other models, making these approaches good performers even in other regression tasks [Cheng et al., 2016,Song et al., 2019].

Further works kept on improving the accuracy, efficiency and latency by: incorporating novel techniques of cross-features importance weighting and mining [Huang et al., 2019,Song et al., 2019,Dilbaz and Saribas, 2023,Liu et al., 2020b,Liu et al., 2020a,Wang et al., 2021b] and more efficient architectures [Bian et al., 2022,Liu et al., 2020a,Cheng et al., 2020,Wang et al., 2021a,Zhang et al., 2023b]. More recently, the works of [Wang et al., 2022a] proposed a novel module to learn context-aware feature representations. Inspired by the strong memory capabilities of Large Language Models in Natural Language Processing, the works of [Zhang and Zhang, 2023] propose to split the learning task accross two networks: a "memory" network responsible for memorizing knowledge about cross features representations effectively, and a "calculator" network responsible for generalization. While the proposed methods do bring improvements and innovative approaches, the state of the art results of [Wang et al., 2023] show that a simple architecture, shown in Figure 2.2, still outperforms the other works given that the feature interaction module efficiently models cross-features and identifies important ones.

The current state of the art models in CTR prediction have made significant improvements in terms of evaluation metrics (area under curve AUC and Log Loss), as shown in the results section of [Wang et al., 2023]. However, these models still suffer from some limitations as their performance tends to decrease as the number of model parameters increases.

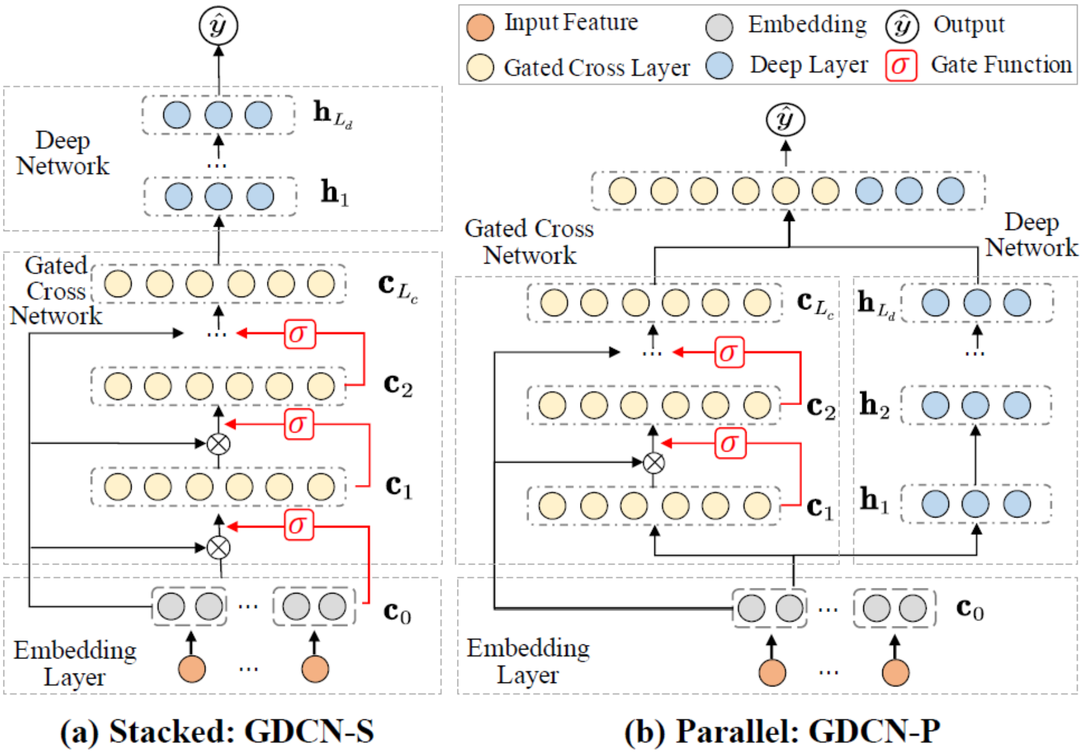



Figure 2.2: The simple architecture of GDCN from [Wang et al., 2023].

### 2.2.2 Real-Time Bidding Strategies

The introduction of Real-Time Bidding (RTB) in 2009 [SETUPAD, 2024] has revolutionized the buying and selling of ad space in the digital advertising world. Introducing an automated, auction-based mechanism that allows advertisers to bid for ad space in real-time as a web page loads. This innovation is fundamental to programmatic advertising, enabling more efficient and targeted ad placements by matching ads with users based on demographic, behavioral, and contextual factors.

The dynamic and competitive nature of RTB requires sophisticated bidding strategies to win the desired auctions while optimizing costs and overall advertising campaigns performance. Therefore, the quest for optimal bidding strategies has become a critical research and development area [Liu et al., 2022, Wang et al., 2017a].

Accurate metric forecasting, particularly click-through rate (CTR) forecasting, is fundamental for the success of a bidding strategy. The ability to predict the likelihood of a user clicking on an ad with precision is pivotal because it estimates the valuation of an impression which directly influences the bid value and the overall strategy. The main framework for bidding strategy is to take into account such metrics forecasts to bid adequately for the ad opportunities with the highest potential return.

The second price auction system, widely adopted in RTB platforms, is a crucial element that significantly affects bidding strategies and the final winning price. In this system, the highest bidder wins the auction but pays the price bid by the second-highest bidder. This mechanism encourages advertisers to bid their true valuation of an impression, theoretically leading to a more efficient market. Nonetheless, it also introduces complexities that are inherent across all auction formats. When strategizing optimal bids, advertisers need to consider their own valuation of an ad impression as well as anticipate how their competitors might value the same impression.

Traditionally, most online ad auctions, especially in programmatic advertising like Google

AdSense, used the second-price auction model. However, there has been a shift towards first-price auctions in recent years [Google, 2024b]. This shift is largely driven by the need for increased transparency in pricing and the desire of publishers to maximize their revenues, which are sometimes perceived to be underrepresented in second-price auctions due to strategic underbidding.

Current bidding strategies fall into two main categories: static and dynamic. Static strategies are defined by their unchanging nature. Once set, these strategies do not adjust to market conditions or variations in data over time. In contrast, dynamic strategies are designed to adapt to changes in the market environment. They continuously analyze incoming data and adjust bids in real-time to optimize performance.

Static strategies, characterized by their simplicity and straightforward implementation, operate under the assumption that the RTB market remains constant or do not account for its inherent fluctuations. Many static strategies, both linear [Perlich et al., 2012, Chen et al., 2011, Liu et al., 2017, Yang et al., 2019] and non-linear [Zhang et al., 2014], have gained widespread adoption across Demand-Side Platforms (DSPs) due to their ease of deployment. Linear bidding strategies employ a direct, proportional relationship between the input variables and the bidding price. This simplicity is what defines linear strategies: the bid amount is calculated as a linear function of one or more variables. Non-linear bidding strategies, in contrast, involve more complex relationships between the variables and the bid price. These strategies can model more intricate dependencies and interactions between features that are not adequately represented by a linear function. Both types of static strategies typically involve fixed or rule-based bidding without adjusting for market changes over time. This rigidity often limits their effectiveness, particularly in dynamic ad delivery environments, resulting in suboptimal campaign performance. Their inability to adjust to new and evolving market conditions can lead to inefficiencies in campaign outcomes.

Contrastingly, dynamic bidding strategies have emerged to address the limitations of static approaches by incorporating the RTB market's volatility into their decision-making

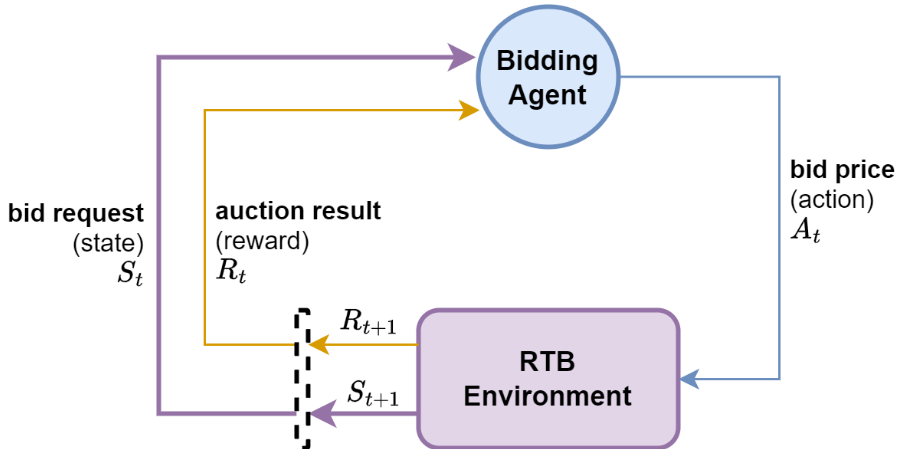



Figure 2.3: Real-Time Bidding modeled as a Markov Decision Process.

processes. Initial approaches such as [Grislain et al., 2019] focused on adapting bids using historical data and predictive modeling. However, the landscape shifted towards RL-based strategies, which offer a more robust framework by conceptualizing the RTB system as an episodic Markov decision process [Du et al., 2017], modeled as shown in Figure 2.3. These strategies promise greater efficiency by dynamically adjusting bids in response to real-time market conditions.

Within the realm of RL-based dynamic bidding strategies, two approaches stand out: single-agent and multi-agent. Single-agent strategies [Cai et al., 2017, Liu et al., 2020c, Wang et al., 2022b, Shih et al., 2023, Du et al., 2017, Wu et al., 2018a, Wang et al., 2017c] focus on optimizing the bidding strategy for individual advertisers, treating the RTB environment and other advertisers as part of the state. The authors in [Du et al., 2017] developed a Constrained Markov Decision Process (CMDP) model for bidding optimization, integrating predicted click-through-rate as the state, bid price as the action, and ad clicks as the reward. Seeking to create a human-level agent, the authors in [Wang et al., 2017c] proposed asynchronous stochastic variant of DQN (Deep Q Network) that uses plain-text descriptions of states from the auctions as inputs to leverage high-level semantic information without complex feature engineering. Model-free approaches were proposed in [Wu et al., 2018a, Liu et al., 2020c]. The works in [Wang et al., 2022b] leverage Bayesian Reinforcement Learning techniques within a curriculum-guided framework to optimize bidding strategies under budget and ROI constraints. Addressing the issue of inaccurate individual predicted CTR, the authors in [Shih et al., 2023] introduced a new evaluation metric, Cluster Expected Win Rate (CEWR) and utilized it to evaluate bid requests by clustering them based on predicted CTRs, ranking the clusters, and setting an Affordability Threshold to allocate budgets. These approaches simplify the modeling process but may not fully capture the interactive dynamics and strategic considerations inherent in the RTB ecosystem.

Multi-Agent RL Strategies, on the other hand, account for the presence and actions of multiple competing and cooperating advertisers within the auction environment [Zhou et al., 2022, Jin et al., 2018, Wen et al., 2022, Guan et al., 2021, Zhao et al., 2018]. This approach acknowledges the complex interplay between different agents, aiming to optimize not only individual performance but also to enhance system-wide outcomes through strategic cooperation and competition. The approach proposed in [Guan et al., 2021] leverages an evolutionary strategy to update network parameters towards Pareto optimal solutions, optimizing multiple objectives simultaneously without compromising others. Seeking the same optimality, the authors in [Zhou et al., 2022] propose the utilization of asynchronous advantage actor-critic (A3C) algorithm to update a global network with different goals.

While most methods have traditionally focused on self-interested optimization for individual advertisers, auto-bidding seeks to harmonize the objectives of increasing platform revenue with optimizing advertisers' revenue, striking a balance between the ecosystem's overall health and individual advertiser success. The works in [Wen et al., 2022] include temperature-regulated credit assignment for mixed cooperative-competitive interaction among agents, and propose bar agents to set a personalized bidding bar for each agent to alleviate the revenue degradation due to the cooperation.

All the bidding strategies share the same principle of relying on forecasted metrics, notably CTR prediction and auction winning price prediction. Accurately predicting

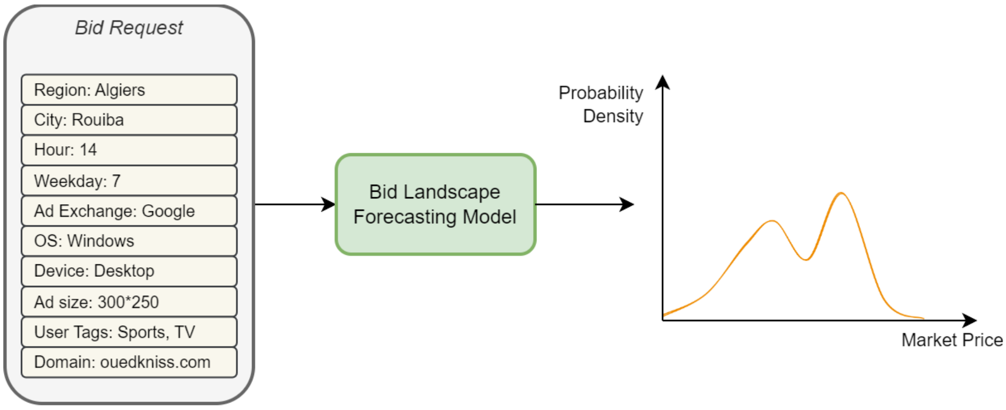



Figure 2.4: Bid Landscape Forecasting or Inventory Pricing.

the winning price, a process known as Inventory Pricing, emerges as a pivotal strategy for success. This prediction not only informs advertisers of the optimal bid to place for an ad impression but also plays a crucial role in enhancing the efficiency of budget allocation across various bidding opportunities. Figure 2.4 illustrates the task.

Some strategies relied on a singular value estimate of the winning price for an ad impression [Wu et al., 2015, Zhou et al., 2021, Zhang et al., 2021b] which usually adapted the models of CTR prediction. While straightforward and easy to interpret, they may not fully capture the variability and uncertainty inherent in the auction environment, potentially leading to either overbidding or underbidding. Therefore, more sophisticated approaches that estimate the distribution of possible winning prices rather than a single point emerged. Early works [Zhang et al., 2014, Zhu et al., 2017, Ren et al., 2018, Wu et al., 2018b] incorporate prior knowledge or assumptions about the distribution of winning prices, which can guide the estimation process and potentially improve accuracy.

However, the complex nature of RTB data often requires more sophisticated modeling approaches beyond traditional well-known distributions to accurately capture the underlying patterns and variability in winning prices. Theoretical distributions often assume independent and linear relationships among variables. However, in RTB environments, the interactions can be highly non-linear and variables are interdependent, influenced by factors like market strategies, budget constraints, and user engagement, which can cause the actual data distribution to deviate significantly from traditional models. Recently, some prior-free models were proposed [Ren et al., 2019, Ghosh et al., 2019, Xi et al., 2021, Li et al., 2022]. Deep Landscape Forecasting [Ren et al., 2019] utilizes Recurrent Neural Networks (RNN) to model the distribution of winning prices by fitting conditional probabilities. This approach has achieved unparalleled performance in capturing complex distribution patterns. The authors in [Li et al., 2022] proposed a novel Neighborhood Likelihood Loss which trains their model to predict the distribution of the winning price by dividing the interval into smooth bins using neighborhood prices and estimates the probability of each neighborhood.

Parallel to winning price prediction, the estimation of click-through rates (CTR) and other relevant metrics' distributions significantly contributes to refining bid strategies. Risk management methods leveraged such distributions estimates to mitigate uncertainties associated with bidding [Zhang et al., 2017, Fan and Delage, 2022, Jiang et al., 2023, Vasile et al., 2017].

Starting from 2017, RTB platforms started to shift towards first price auctions [SmartyAds, 2017, Numberly, 2024] where the highest bidder pays the price they bid, contrary to the second price auction's bid-and-pay-the-second-highest-price mechanism. This change required either the design of new bidding strategies or the adaptation of already existing successful bidding strategies. Many works focused on adapting existing methods using bid shading techniques [Zhang et al., 2021b, Zhou et al., 2021]. Bid shading enables advertisers to strategically lower their bids to just the right level where they can win the auction without overpaying, ensuring cost-effectiveness while maintaining competitiveness. In second-price auctions, bidders traditionally bid their maximum because they pay the second-highest price, mitigating overpayment. However, in first-price auctions, this strategy leads to overpaying. Bid shading helps RTB agents adapt by slightly lowering their bids to avoid overpayment while still remaining competitive. This strategy involves understanding market dynamics and adjusting bids to reflect true market value, essential in the first-price environment. The authors in [Zhou et al., 2021] propose a deep distribution network to learn the distribution of minimum winning price for both censored and non-censored first price auctions. The works in [Zhang et al., 2021b] leverages exponential weighting to perform dynamic binning via subsequent splitting and merging operations based on incoming data, updating candidate bidding prices periodically, and incorporating a discount factor for adaptive learning.

Multi-stage bid optimization acknowledges the sequential and dependent nature of bidding decisions within a campaign. Unlike models that treat each bid as an independent event, multi-stage strategies consider how previous bids, outcomes, and remaining budget impact future bidding decisions. Integral to the concept of multi-stage optimization is the evolution of budget pacing methods [Lee et al., 2013, Xu et al., 2015, Wu et al., 2018a, Jiang et al., 2023]. These strategies are designed to optimally distribute the campaign budget across its lifespan, ensuring that spending is aligned with strategic goals at various stages. Early methods for budget pacing primarily depended on linear programming or control feedback loops approaches, which provided structured yet rigid frameworks for managing campaign budgets. In contrast, the most effective methods today increasingly integrate budget pacing into RL models.

Reinforcement Learning methods in Real-Time Bidding offer significant advancements in bid optimization and campaign management. However, they also come with certain drawbacks that can impact their implementation and effectiveness in practical scenarios : complexity in model design, data and computational demands, exploration-exploitation trade-off, real-time adaptability, generalization accross campaigns and market conditions. The authors in [Mou et al., 2022] interestingly addresses the inconsistency between online and offline environments in auto-bidding systems, which degrades the performance of RL-based auto-bidding policies. They introduce a Sustainable Online RL (SORL) framework for training auto-bidding policies directly with the real-world advertising system (RAS), including a Safe and Efficient online exploration (SER) policy and a Variance-suppressed Conservative Q-learning (V-CQL) method for offline training.

### 2.2.3 Advertising Strategy & Bundle Recommendation

An advertising strategy is a combination of audience segments, ad placements, geographical targets, etc. with the ultimate goal of achieving optimal engagement and conversion rates. The optimization of such targetings and configurations is extremely important to improve the overall campaign performance. It involves fine-tuning the selection of audience demographics, content placement, and scheduling. Such optimization not only ensures that marketing messages reach the right audiences at the right time and place but also maximizes return on investment by reducing wasted ad spend on less interested or irrelevant segments.

In Section 2.2.1, the focus was on learning how to estimate the utility of an advertising opportunity. In Section 2.2.2, the focus was on leveraging the utility estimates to learn how to act when an advertising opportunity presents itself. In practice, not all ad opportunities are presented to an advertiser. The advertiser has to first define a set of advertising strategies as a subset of targetings and campaign settings. If an ad opportunity matches an advertising strategy's attributes, it will be presented to the bidding agent to act on it. Ideally, an advertising strategy should be designed such that it will filter out all the non-interesting ad opportunities and keep only the most promising ones in terms of ROI maximization likelihood [Guo et al., 2021,Miralles-Pechuán et al., 2023].

In this section, we explore the advertising strategy recommendation problem. Which, despite its obvious significance, remains underexplored in research due to its high complexity and challenging constraints.

Traditionally, recommender systems, pivotal in shaping online user experiences, were primarily designed to suggest individual items to users. Whether for movies, books, or products, these systems aimed to predict and propose items that a user is likely to appreciate or engage with, based on their past behavior and preferences. These systems typically predict the utility of each item independently then perform a ranking task to propose a list of independent items to users. This focus on single-item recommendation is deeply rooted in the initial objectives of recommender systems, which were to simplify and personalize the online content discovery process for users. Traditional methods of recommender systems have been widely used in many candidate generation problems due to their capacity to estimate the utility of any given candidate and capture some level of feature interactions [Wermser et al., 2011a, Covington et al., 2016b, Frolov and Oseledets, 2017, Xue et al., 2017, He et al., 2017].

Unlike conventional recommender systems that suggest ranked lists of independent items, advertising strategy recommendation involves proposing a combination of interdependent items. These can include a mix of advertising mediums, ad images, promotional messages, target audience segments, and timing configurations, all tailored to collectively optimize the effectiveness of a marketing campaign. This complexity arises from the interdependent nature of campaign components, where the success of the strategy hinges on the synergistic performance of its parts, rather than the impact of isolated items.

This naturally leads to the concept of the bundle recommendation problem. Bundle recommendation focuses on suggesting a set of items that are expected to work well together, enhancing the overall utility. In the realm of advertising strategy design, this translates into identifying and recommending a cohesive combination of targetings and configurations of campaign elements that drive towards marketing objectives. The distinctiveness of the bundle recommendation problem lies in its focus on the compatibility and interdependence of the items within the bundle. It underscores the idea that the value derived from the bundle is contingent upon the entire combination of its components functioning as a unified whole, rather than being attributed to any individual part or a subset of parts [Sar Shalom et al., 2016, Pathak et al., 2017].

The advertising strategy recommendation problem represents a specific instance within the wider spectrum of the bundle recommendation task, characterized by predetermined bundle sizes and fixed item categories.

Early attempts to tackle this complex problem employed methods such as integer programming or constraint solvers [Marchetti-Spaccamela and Vercellis, 1995, Xie et al., 2010, Zhu et al., 2014]. These mathematical approaches set the stage for more nuanced and specialized techniques to emerge in the pursuit of effective bundle recommendations. However, they often struggle with scalability due to the exponential increase in computational complexity as the problem size grows, making it impractical for large-scale applications.

Further advancements in the field saw the application of association rule mining and bundle mining approaches [Fang et al., 2018, Beheshtian-Ardakani et al., 2018]. These methods delve into patterns of item associations within large datasets, aiming to uncover frequent item combinations suggesting a natural inclination for co-occurrence. Such methods might produce an overwhelming number of rules, many of which may be irrelevant or trivial, leading to challenges in filtering and prioritizing the most significant associations.

In the same spirit, evolutionary algorithms, such as genetic algorithms [Miralles-Pechuán et al., 2018, Ying and Zhizhong, 2009], emerged as a potent form of bundle mining due to their ability to handle combinatorial optimization problems. However, these methods suffer from slow convergence rates, the risk of premature convergence to local optima, and the inability to control exploration and exploitation rates in an efficient manner.

Building upon the success of traditional single-item recommender systems, some researchers explored the use of tensor factorization techniques, including factorization machines, for multi-item recommendation [Wermser et al., 2011b, Chen et al., 2017, Hong and Jung, 2018]. This approach extends the concept of matrix factorization by incorporating multiple dimensions, illustrated in Figure 2.5, allowing for a richer representation of interactions between users, items, and additional context, which is crucial for effective bundle recommendation. LIRE [Liu et al., 2014] and BBPR [Pathak et al., 2017] train Bayesian ranking models to simultaneously learn user preference towards items and bundles. BBPR can further generate new bundles using a greedy annealing sched-

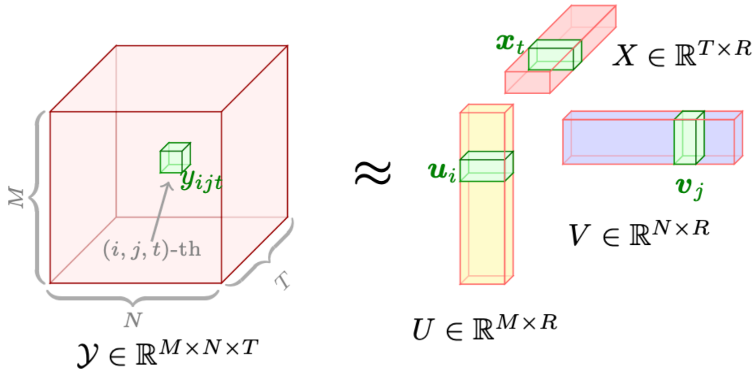



Figure 2.5: Tensor Decomposition.

ule. EFM [Cao et al., 2017] jointly factorizes user-item, user-bundle interaction matrices and item-item-bundle co-occurrence matrices, to capture user preference over items and bundles. While quite effective, such methods may require a significant amount of training data to capture higher-order interactions, which can be a barrier. Moreover, they still require the calculation of each possible combination, which leads to a combinatorial explosion problem.

Some works have employed multi-facet Multi-Armed Bandits to understand the distribution of bundle utility and item effectiveness [Li et al., 2010, Qin et al., 2014]. These methods use statistical models to learn from interactions and employ sampling techniques to select multiple items, balancing the exploration of new item combinations with the exploitation of known, effective bundles. As shown in Figure 2.6, these methods model each strategy feature as a multi-armed bandit in a framework of bandits that collectively model a single final reward. The proposed approach in [Ban et al., 2021] enhances reward optimization by integrating neural networks with a UCB strategy in a multi-facet bandit framework for improved arm selection. BYOB [Deng et al., 2021] formulates the problem as a combinatorial optimization problem over a set of candidate items and applies a policy-based deep reinforcement learning algorithm to solve it. While effective in balancing exploration and exploitation and capturing complex item relationships, multi-armed bandits models are not specifically designed for bundle gen-

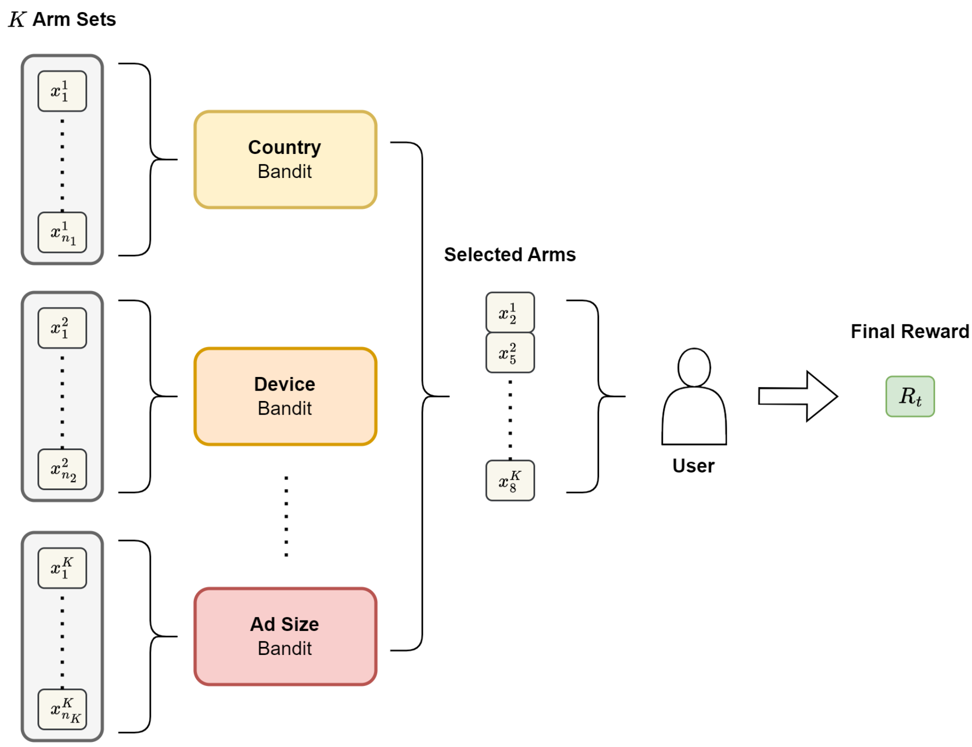



Figure 2.6: Multi-facet Multi-Armed Bandits modeling an advertising strategy final reward.

eration as they only model the independant distributions of items. Making the sampling techniques for bundle generation sub-optimal by definition.

Graph Neural Networks (GNNs) and graph-building techniques have also been pivotal in advancing bundle recommendation systems [Ying et al., 2018, Gong et al., 2019, Wang et al., 2019, Zhang et al., 2022, Li et al., 2023, Ma et al., 2024, Liu et al., 2023b, Wei et al., 2023]. By modeling the items and their relationships as a graph, these approaches leverage the power of GNNs to capture complex interactions and dependencies between items. BGCN [Chang et al., 2020, Chang et al., 2023] unifies user-item, user-bundle interactions and bundle-item affinity into a heterogeneous graph, and adopts Graph Convolutional Network (GCN) [Wang et al., 2019] to perform item- and bundle-level propagation to learn user and bundle representations with item level semantics. GRAM-SMOT [Vijaikumar et al., 2020] utilizes a graph attention-based framework to learn higher-order relationships between users, items, and bundles. It incorporates a loss function based on metric-learning to efficiently learn entity embeddings. To generate novel bundles, the model leverages sub-modular function maximization. BundleNet [Deng et al., 2020] applies GCN on the user-item-bundle tripartite graph and formalized the bundle recommendation problem as a link prediction problem. Graph representations are easy to interpret, usually making the translation process of a problem into a graph structure quite straightforward. However, graph-based models struggle from some limitations such as requiring a substantial amount of data to accurately model the nodes and connections, their performance falls off with the increase of graph size, and they can be resource intensive.

In addition to viewing bundle recommendation as a set generation task, some research has approached it as an item sequence generation task [Beutel et al., 2018, Hu and He, 2019, Sun et al., 2019, Katz et al., 2022], which, despite being suboptimal by definition, simplifies the problem sufficiently to yield satisfactory results. Figure 2.7 illustrates an example of such approach. BGN [Bai et al., 2019] adopts a sequence generation model and integrates masked beam search to produce high-quality, diversified bundles. PoG [Chen et al., 2019b] adopts an encoder-decoder transformer based framework to generate multiple items as a personalized bundle. Analogous to the bundle generation task, VTN [Arroyo et al., 2021] utilizes self-attention layers in a variational autoencoder (VAE) framework to capture relationships between elements in a layout and showcases both an auto-regressive and a non-autoregressive decoding modes during the generation process. The proposed method in [Bibas et al., 2023] involves maintaining a latent space for each item category and translating item representations into these category spaces to provide suitable recommendations. The model incorporates ideas from Cycle Generative Adversarial Networks to generate the next item.

Sequence models, especially when treated as item sequence generation problems, might not inherently capture the set-based nature of some recommendation scenarios. Their performance can degrade when the order of items in the recommendation is not inher-

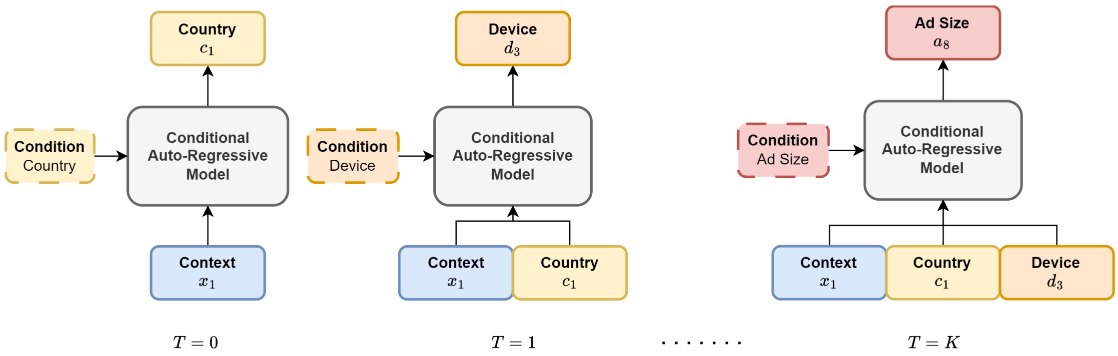



Figure 2.7: Conditional Auto-Regressive Generative Model.

ently meaningful, potentially leading to suboptimal bundle configurations.

The integration of the attention mechanism [Vaswani et al., 2017] into bundle generation models has further improved their ability to generate relevant bundles while capturing complex relationships between items within a bundle [Chen et al., 2019a, He et al., 2019, Sun et al., 2019, Arroyo et al., 2021, He et al., 2022, Wei et al., 2023]. These models can dynamically assign weights reflecting the importance of different items, taking into account the context and the specific characteristics of each item to improve recommendation quality. BRUCE [Brosh et al., 2022] adapts transformers to represent user, item, and bundle data. It utilizes the self-attention mechanism to capture latent relations between items in a bundle, users' preferences towards items, and the entire bundle. It then aggregates the outputs of the transformer into a prediction layer to score the affinity between the user and the bundle. The generation process consists of iterating through candidate bundles and ranking them. More similar to our work, Conna [Wei et al., 2022a] includes a type-aware encoder to learn representations for different candidate items and a non-autoregressive decoder that generates all the items of the bundle in one pass. The model is optimized via constrastive learning to further enhance the quality of the generated bundles. Its major novelty is that it doesn't iterate through candidate bundles to rank them, nor does it generate the items one by one, significantly reducing the computational demands associated with bundle generation.

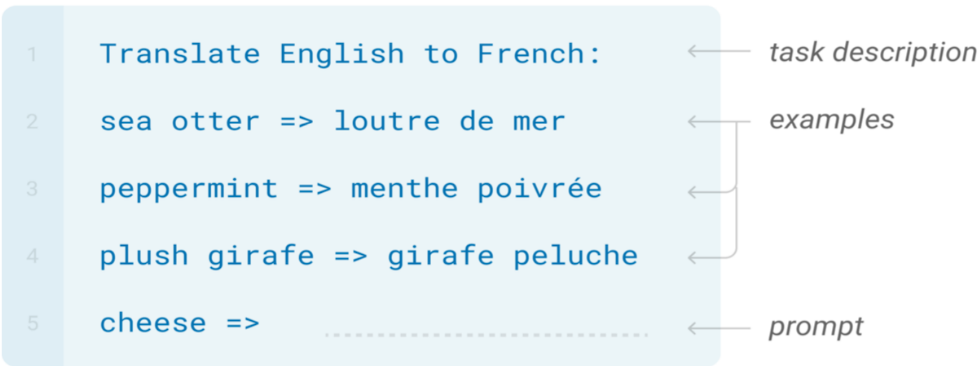



Figure 2.8: Example of LLM in-context learning.

Most recently, the rise of Large Language Models (LLMs) has opened new frontiers for multiple generative tasks. Leveraging their ability to understand and generate complex patterns of data, some works have used LLMs to directly generate bundles of items [Zhu et al., 2023, Harte et al., 2023, Zhang et al., 2023a, Dai et al., 2023, Sanner et al., 2023]. This approach capitalizes on the deep contextual understanding and generative capabilities of LLMs. In-context learning (ICL) combined with prompt engineering is one of the most used approaches to tailor LLMs for a specific purpose by providing the model with examples or prompts that are pertinent to the task at hand, as shown in Figure 2.8, thereby giving it the necessary context to guide its outputs. Alternatively, when a few examples are not sufficient to provide a rich context, another approach involves fine-tuning LLMs by further training the model on a task-specific dataset, therefore aligning its capabilities more closely with the requirements of the task [Bao et al., 2023, Yin et al., 2023, Chen, 2023].

Even though Large Language Models (LLMs) showcase versatility and strong performance (refer to Section 2.2.4.1), they struggle to provide strictly structured recommendations. Their fundamental design is sequential, which hampers their ability to consistently enforce a structured output. While these models attempt to approximate structured outputs, the results are often unreliable, leading to suboptimal outcomes in scenarios requiring precise structure.

Despite the diversity in methodologies for bundle generation, a prevalent issue among most is their lack of scalability to the vastness of web-scale applications such as advertising strategy generation. They either struggle with the challenges posed by combinatorial explosion, inherent high complexity, or they are deliberately designed to accept sub-optimality as a trade-off for manageability. This limitation significantly hinders their applicability in real-world advertising contexts. The consequence is a gap between theoretical models and practical, scalable applications, highlighting a crucial area for further research and innovation in advertising strategy recommendation systems.

### 2.2.4 Generative Models & Impactful Concepts

#### 2.2.4.1 The Recent Success of Generative AI

Generative models, particularly text and image generation models, have seen a meteoric rise in recent years [OpenAI, 2023, Touvron et al., 2023, Abdullah et al., 2022, Peebles and Xie, 2023, Rombach et al., 2022a]. These models are leading a big change in how content is created, thanks to their ability generate engaging and personalized content on a large scale in a fraction of the time it would take human creators.

Central to this transformation are generative models like Generative Adversarial Networks (GANs) and diffusion-based models for images [Rombach et al., 2022a, Peebles and Xie, 2023, Betker et al., 2023, Ramesh et al., 2021, Dhariwal and Nichol, 2021, Chang et al., 2022, Esser et al., 2021, Saharia et al., 2022] and Transformer-based models for text [OpenAI, 2023, Brown et al., 2020, Myers et al., 2024]. These models have demonstrated remarkable capability in producing outputs of high quality and diversity. By learning to replicate the distribution of data in the real world, they are capable of generating content that is often indistinguishable from human created content.

Several studies have focused on leveraging the capabilities of these models in the field of digital advertising to improve advertising effectiveness. These applications primarily fall into two categories: generating visually compelling ad creatives and banners [Ku et al., 2023, Lin et al., 2023, Vaddamanu et al., 2022, Wei et al., 2022a], and the generation of convincing advertising texts and snippets [Thomaidou et al., 2013, Hughes et al., 2019, Wei et al., 2022b, Yao et al., 2023, Kamigaito et al., 2021], therefore significantly boosting the impact and attractiveness of digital advertising campaigns.

While the exploration of text and image generative models in digital advertising is not the core focus of this thesis, the advanced machine learning techniques and concepts that power these models (such as attention mechanisms, variational generative methods, and contrastive learning) have proven to be incredibly powerful and their value extends beyond the scope of any singular field. In the next sections, we provide an overview of the key concepts and methods that held substantial importance throughout the course of this thesis.

#### 2.2.4.2 Attention Mechanism

The attention mechanism revolutionized the way models process sequences, by allowing them to dynamically assign weights reflecting the importance of different parts of the input data [Vaswani et al., 2017, Brauwers and Frasincar, 2023, Devlin et al., 2019]. The foundation of the attention mechanism's success lies in the architecture of transformers, shown in Figure 2.9. This innovation led to substantial improvements in tasks requiring understanding of context and relationships within data, such as language translation and image processing. Recent efforts have sought to address some of the original transformer model's limitations [Gu et al., 2022, Smith et al., 2023, Gu and Dao, 2023, Bulatov et al., 2023], leading to variations like the Longformer [Beltagy et al., 2020], designed for handling longer sequences, or Autoformer [Wu et al., 2021] for time series forecasting, or even vision [Dosovitskiy et al., 2021]. These successful adaptations highlight the enhancements that the attention mechanism has brought to a wide array of tasks.

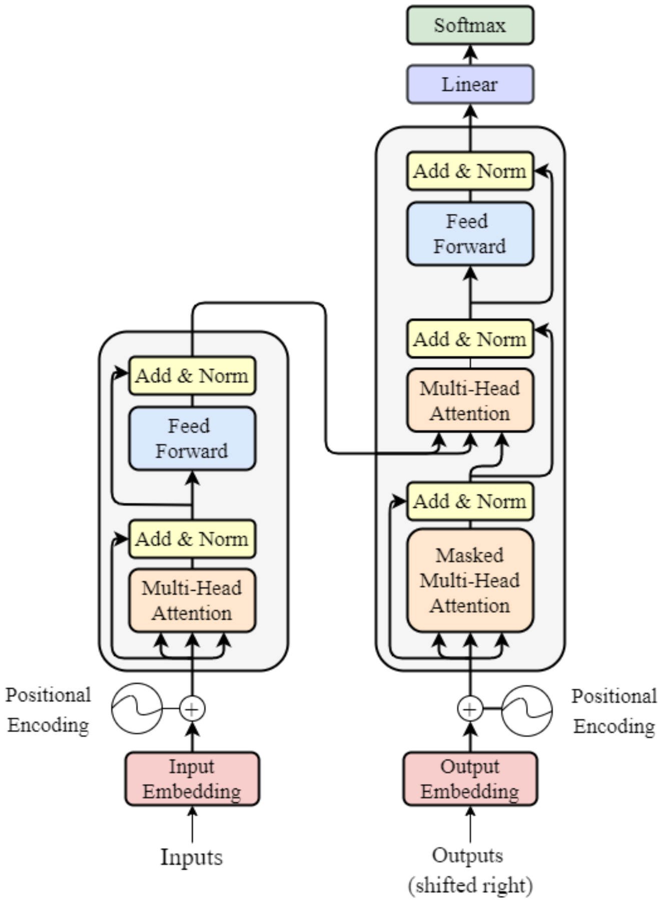



Figure 2.9: The famous Transformer architecture from [Vaswani et al., 2017].

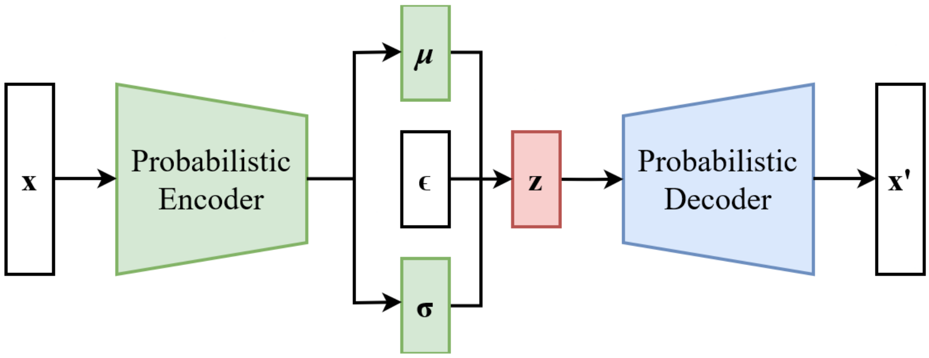



Figure 2.10: The architecture of Variational Auto-Encoder.

#### 2.2.4.3 Variational Methods

Variational methods enable models to generate new data instances by learning the complex distribution of input data transforming it into a form where sampling is theoretically well-founded and practically executable. Variational Auto-Encoders (VAEs) [Kingma and Welling, 2014], shown in Figure 2.10, leverage the principles of Bayesian inference to learn the underlying probability distribution of input data. The encoder transforms input data into a latent space representation, effectively compressing the data into a condensed form, while the decoder reconstructs data from this latent representation, aiming to generate outputs that mirror the original inputs. The latent representation is restricted to a tractable distribution, typically a Gaussian distribution, implying that in the generation phase, sampling occurs from this Gaussian distribution, followed by the employment of the decoder to produce a new data instance. Numerous other works incorporate this concept into their models to imbue them with variational characteristics [Arroyo et al., 2021, Hu et al., 2022, Fang et al., 2021, Lin et al., 2020b].

Generative adversarial networks (GANs) [Goodfellow et al., 2014, Karras et al., 2019], Latent Diffusion models [Rombach et al., 2022b, Dhariwal and Nichol, 2021] and Normalizing flows [Rezende and Mohamed, 2015, Kingma and Dhariwal, 2018] all originate from the same starting point, utilizing a distribution of noise, notably Gaussian noise, as

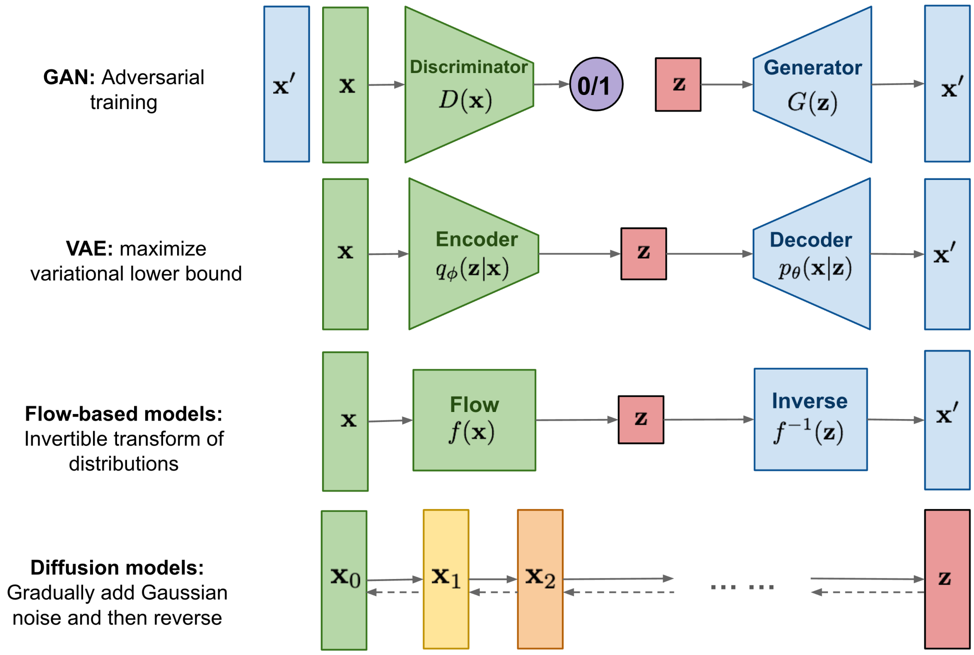



Figure 2.11: Variational Generative Models: GANs, VAEs, Normalizing Flows, Diffusion Models.

their foundation to either directly generate an output or progressively diminish the noise through iteration. Figure 2.11 shows the various architectures of variational generative approaches.

Recently, vector quantization has emerged as a powerful technique, particularly in models like VQ-VAE [van den Oord et al., 2017, Razavi et al., 2019, Lancucki et al., 2020], for generating high-fidelity results, leading to innovations in variational methods that further enhance performance and versatility. Vector Quantization, illustrated in Figure 2.12, is a technique that involves mapping input data to a finite set of vectors in a way that quantizes the input space. VQ-VAE [van den Oord et al., 2017], shown in Figure 2.13, introduced the initial formulation, including a commitment loss and Exponential Moving Averages for improved codebook learning. VQ-GAN [Esser et al., 2021, Yu et al., 2022], shown in Figure 2.14, used VQ-VAE in its auto-encoder module and learns to decode from a quantized representation. It later trains a transformer-like model to learn how to

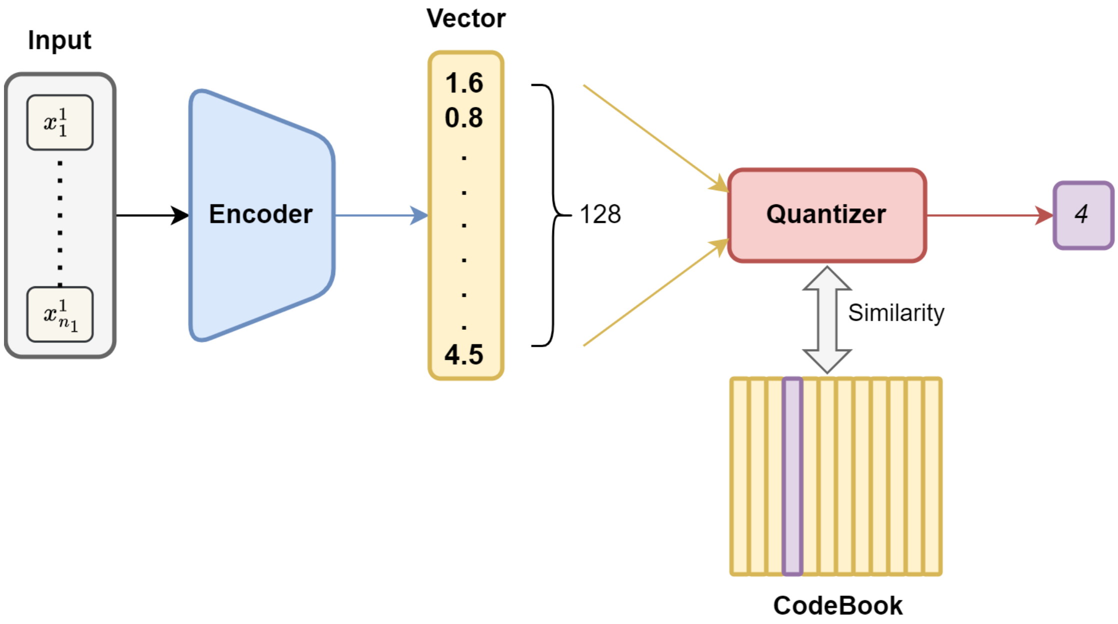



Figure 2.12: Illustration of the quantization process.

generate new quantized representations, then uses the decoder to generate a new data instance. Approaches such as [Kolesnikov et al., 2022, Chang et al., 2022] leverage the same technique with an auto-regressive and a masked generation method respectively, exemplified in Figure 2.15.

Subsequent variants of vector quantization have been developed to either enhance performance or address issues associated with the codebook. FSQ [Mentzer et al., 2023] employed bounded scalar quantization, which quantizes codes using scalars rather than real numbers, by truncating them to the nearest scalar value. RQ-VAE and RQ-Transformer in [Lee et al., 2022] used residual quantization, where quantized codes are refined by additionally storing (quantized) residuals, as shown in Figure 2.16. The authors in [El-Nouby et al., 2023] proposed product quantization, where the codebook is factored into a product of smaller codebooks.

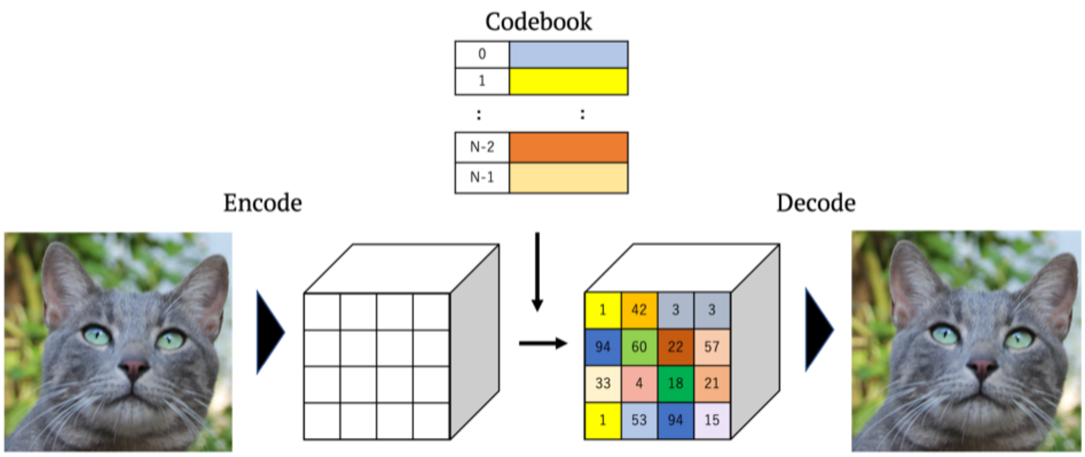



Figure 2.13: Simplified architecture of VQ-VAE.

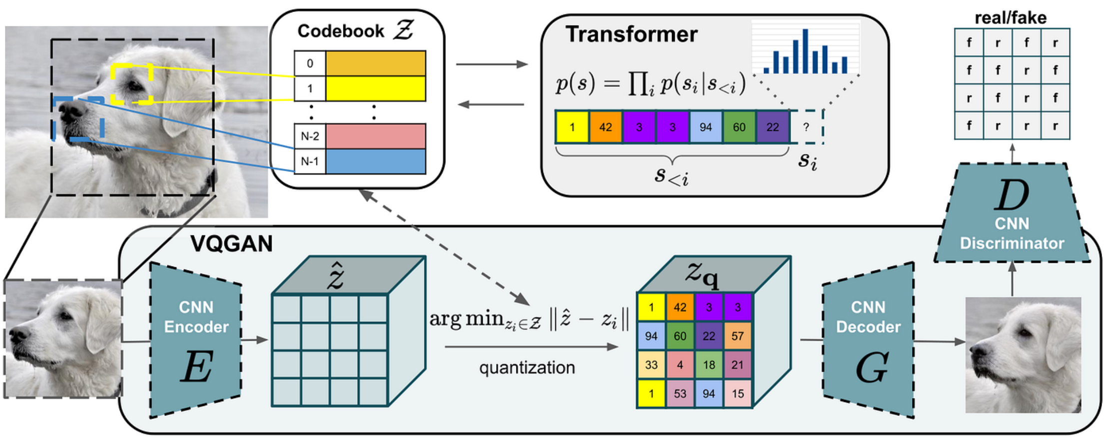
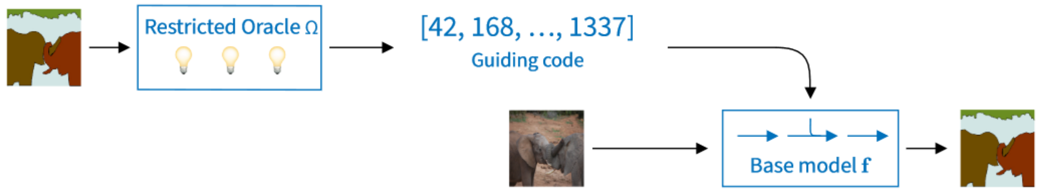
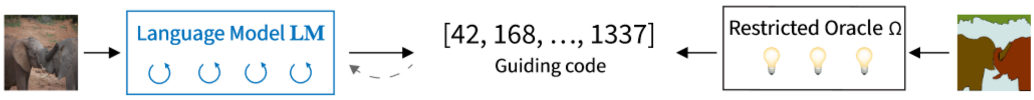



Figure 2.14: Architecture of VQ-GAN.




(a) **Stage I** training: we train the base model $f$, which is guided by the code produced by the *restricted oracle* model $\Omega$. The oracle has access to the ground-truth label, but is only allowed to communicate with $f$ by passing a short discrete sequence, which we call a *guiding code*.




(b) **Stage II** training: we train a *language model* (LM) to output a *guiding code* by learning to mimic the oracle, but using only the image input.

Figure 2.15: Auto-regressive quantization based generation of UViM [Kolesnikov et al., 2022].

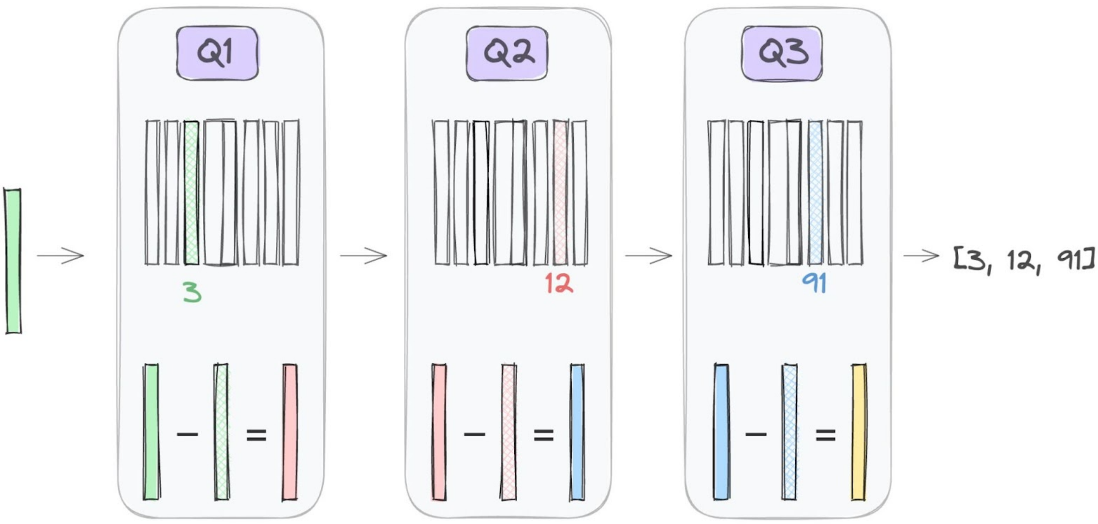



Figure 2.16: Illustration of Residual Quantization.

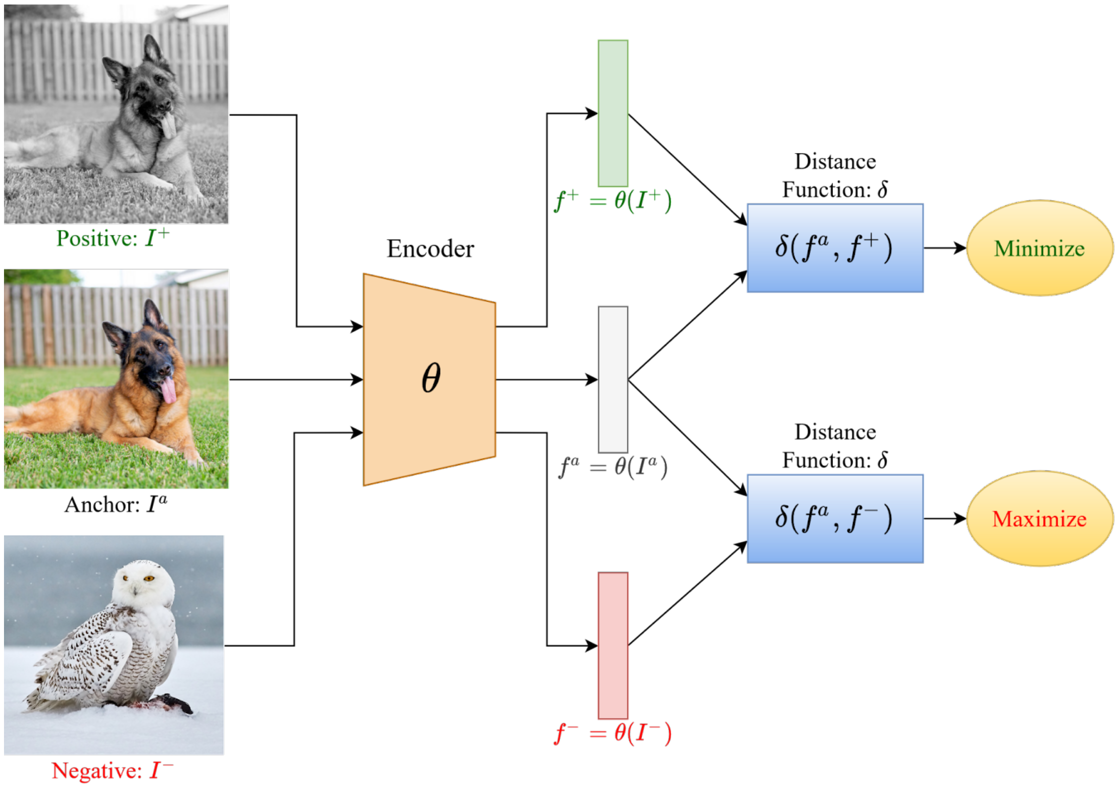



Figure 2.17: Contrastive Learning.

#### 2.2.4.4 Contrastive Learning

Contrastive learning [Khosla et al., 2020] is a technique in unsupervised or semi-supervised machine learning that aims to learn effective representations by contrasting positive examples against negative examples. Positive examples are typically pairs of data points that are considered similar or related, while negative examples are those deemed dissimilar or unrelated. The objective of this learning paradigm is to adjust the model's parameters such that representations of positive pairs are brought closer in the latent space, whereas those of negative pairs are pushed apart. Figure 2.17 illustrates an example of contrastive learning.

The ability of contrastive learning to refine feature representations makes it particularly valuable in tasks requiring a high degree of specificity and relevance, such as personalized advertising. By ensuring that the generated content is distinct and directly relevant to the target demographic, contrastive learning enhances model performance therefore improving the overall impact and effectiveness of advertising campaigns.

#### 2.2.4.5 Metric Learning

When evaluating complex data types, such as images, where the data cannot be reduced to simple scalars, applying conventional metrics like Mean Squared Error or Cross Entropy becomes less straightforward and may not accurately capture the essence of the evaluation objectives. In contrast to these traditional metrics, learned metrics leverage machine learning models to gauge the quality of generated data [Kaya and Bilge, 2019, Sung et al., 2018, Radford et al., 2021]. This approach ensures that the evaluation criteria are closely tailored to the specific nuances and intricacies of the given task, offering a more precise and meaningful assessment of data quality.

Typically, evaluation criteria can be conceptualized in human terms; for instance, we might intuitively understand that a certain output should yield a high metric value, while another should register low. However, encapsulating such nuanced criteria using conventional metrics often proves challenging. To bridge this gap, a machine learning model can be trained to emulate human evaluative behavior, thereby translating it into a differentiable function. This approach enables the formalization of subjective assessment criteria into quantifiable measures that can guide model training and refine the evaluation process. CLIP [Radford et al., 2021] measures the similarity between text and image pairs and serves as the backbone of the famous text-to-image generation model DALL-E [Ramesh et al., 2022].

A notable architectural feature of learned metrics is the incorporation of a neural network, often initiating with an embedding layer. This configuration, while effective in mapping inputs to a richly descriptive high-dimensional space, introduces a critical challenge: the disruption of gradient flow from the metric back to the model in-training.

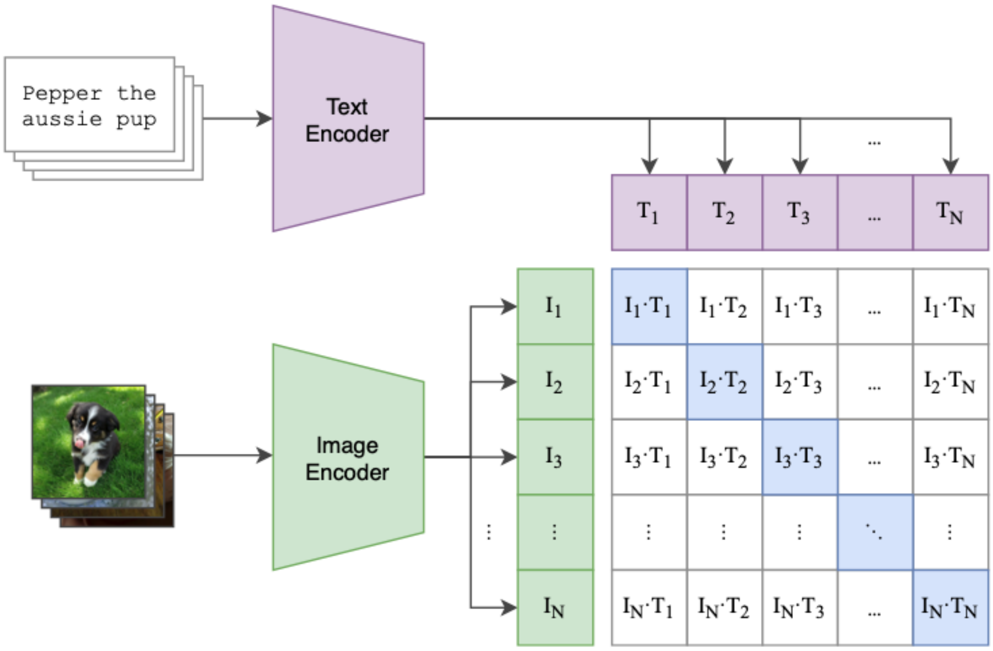



Figure 2.18: CLIP [Radford et al., 2021] leverages contrastive learning to learn a similarity metric between texts and images.

This disruption arises due to the embedding layer, which impedes the backward propagation of gradients, essential for model training based on the feedback provided by the learned metrics.

Straight-Through Estimators [Bengio et al., 2013], a technique heralded for its utility in VQ-VAE [van den Oord et al., 2017], facilitate the bypassing of gradients through non-differentiable operations, thereby reinstating the continuity of gradient flow. Figure 2.19 illustrates the gradients bypass. When this approach is combined with Gumbel-Softmax sampling [Jang et al., 2017], a method enabling differentiable sampling from discrete distributions, it not only mitigates the issue of disrupted gradient flow but also enhances the model's capacity to produce diverse and relevant outputs.

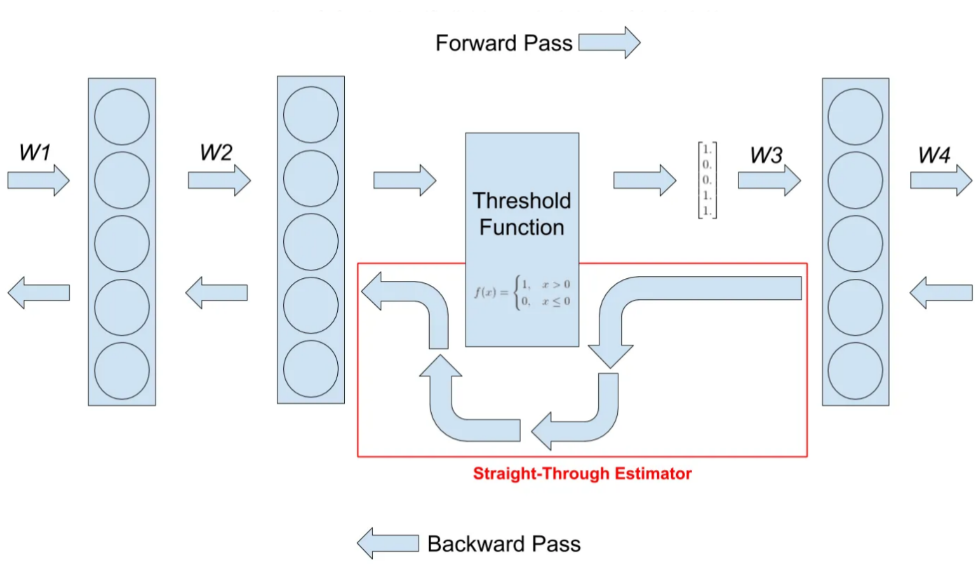



Figure 2.19: Straight-Through Estimator method: gradients are approximated and passed directly back to previous layers.

#### 2.2.4.6 Controllable Generation

Generative models have marked a significant advancement in the field of artificial intelligence, demonstrating remarkable ability in creating realistic and diverse outputs across various domains such as text, images, and music. However, alongside the appreciation for these models' capabilities, there exists a pronounced aspiration for mechanisms to exert control over the generative process. This desire stems from several considerations, ranging from the need for guidance and specificity in the generated outputs to demands for contextual personalization and adherence to ethical standards.

The ability to direct the output of generative models holds immense value. For instance, in creative domains, it enables artists and designers to generate content that aligns closely with their vision by specifying certain attributes or styles. In marketing, controllable generation allows for the creation of tailored advertisements that resonate with distinct audience segments, enhancing engagement and effectiveness. Furthermore, from an ethical perspective, control mechanisms can help mitigate the generation of harmful

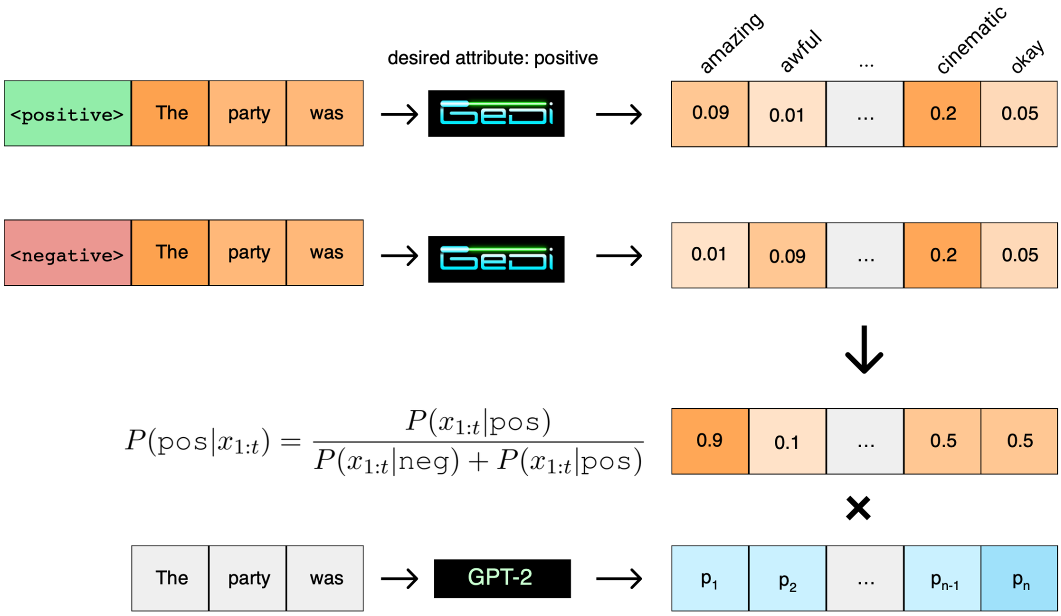



Figure 2.20: Text Conditional Generation via sentiment labels.

or biased content, ensuring that outputs conform to societal norms and values.

Several methodologies have been developed to introduce controllability into generative models, each offering different mechanisms for influencing the generation process:

**Conditional/Contextual Input:** This approach involves modifying the model to accept additional input that specifies desired attributes of the output. For example, in image generation, a model could take a textual description as input to generate an image that matches the description [Nichol et al., 2022, Ramesh et al., 2022, Patashnik et al., 2021]. In text generation, a sentiment label could steer the model to produce content with a specified emotional tone, as shown in Figure 2.20. This approach is readily adaptable to pre-existing generative models [Mirza and Osindero, 2014, Zhang et al., 2021a, Ilias and Askounis, 2023]. For larger models, where comprehensive retraining is prohibitively expensive, certain techniques allow for fine-tuning on new concepts to enable conditioning, providing an efficient means to augment them with controllable capabilities [Dong et al., 2022, Ruiz et al., 2023].

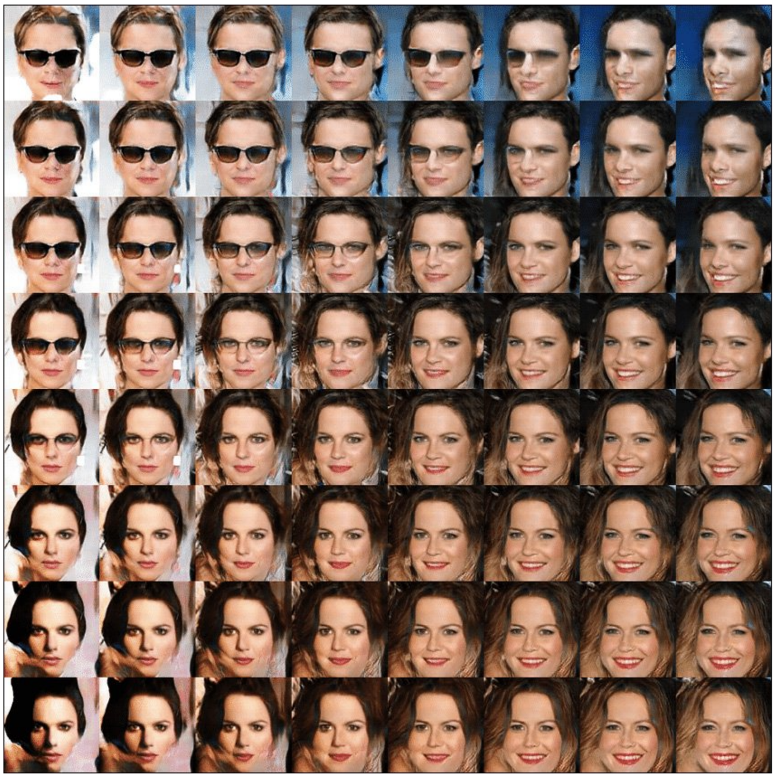


Figure 2.21: Latent Space Manipulation from [Pieters and Wiering, 2018].

**Latent Space Manipulation:** Given that many models compress input data into a latent space it becomes possible to influence the characteristics of the generated outputs by strategically manipulating points within this space [Kingma and Welling, 2014, Goodfellow et al., 2014, Roberts et al., 2018]. In Figure 2.21, it is shown how an interpolation between various points in the latent space affects the generated output. However, effectively leveraging this technique necessitates a comprehensive grasp of the latent space's structure and its relationship with the attributes of the output.

**Attention Manipulation:** For models that employ attention mechanisms, such as Transformers [Vaswani et al., 2017], controlling the focus of the attention layers can guide the model to prioritize certain aspects of the input, thereby influencing the characteristics of the output. The works in [Hertz et al., 2023, Tumanyan et al., 2023], shown in Figure 2.22, manipulate cross-attention or spatial features weights to edit both global and local

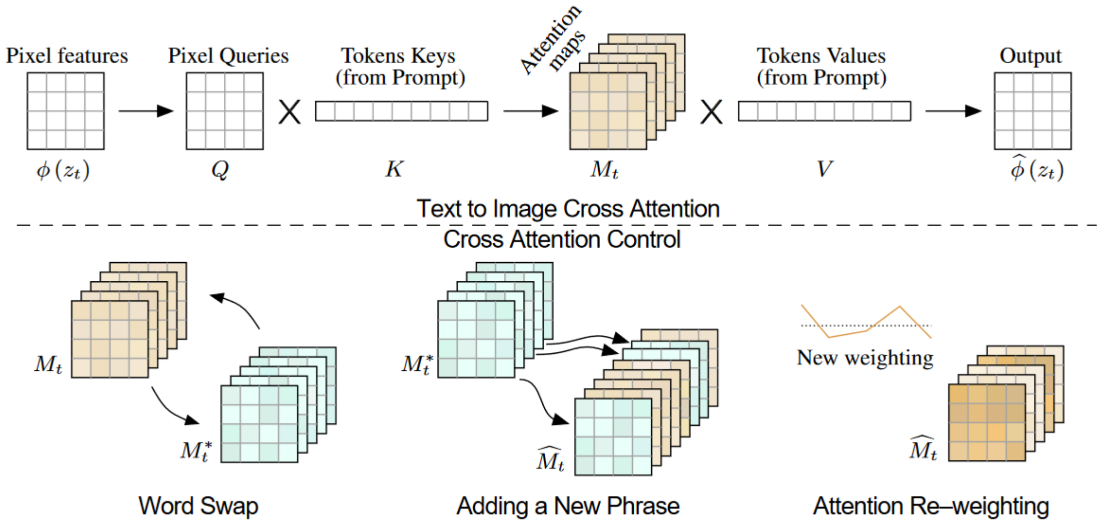



Figure 2.22: Method overview of Attention Manipulation in [Hertz et al., 2023].

aspects of the image by changing the text prompt directly, but they tend to preserve the original layout of the source image and fail to handle non-rigid transformations. The methods in [Tewel et al., 2023, Kumari et al., 2023, Cao et al., 2023] delve further by adjusting the Q (query) and K (key) components of the attention mechanism for more profound control.

## 2.3 Our Positioning & Summary

The primary emphasis of this thesis is on advertising strategy recommendation/generation, an area that remains relatively underexplored within digital advertising research. In the initial stages of this thesis, we delved into CTR prediction and RTB strategies as foundational elements. However, it became apparent that the methods proposed in these areas, while effective for their specific purposes, do not align well with our goals. This discrepancy is rooted in the challenges of exploring advertising strategies in a non-combinatorial manner and the imperative to avoid clear inefficiencies, such as using autoregressive methods for data that does not inherently possess a sequential order.

Consequently, our contributions navigate through largely uncharted waters in digital advertising. Nonetheless, significant parallels can be drawn to the task of bundle recommendation, which has served as a substantial source of inspiration for our work.

Table 2.1 provides a comprehensive summary of the primary methods investigated in this chapter, outlining their overall strengths and weaknesses, and Table 2.2 provides references for the methods. Table 2.3 distills this information and focuses on presenting a concise comparison that illustrates the positioning of our contributions in relation to existing methods.

Our contributions stand out due to our thorough approach to addressing the specific challenges and constraints unique to advertising strategy recommendation. Key among these challenges are the exploration of the advertising strategy feature space in a non-combinatorial manner, the ability to handle order-agnostic modeling without depending on less efficient autoregressive techniques, and the capability to modulate generative fidelity to training data to optimally balance between exploration and exploitation as needed. Furthermore, our methods adeptly model intricate feature interactions and provide versatile control over the generation process, setting our work apart in the field of digital advertising.

Table 2.1: Summary of State of the Art methods in Machine Learning Applied to Digital Advertising

| Task | Category | Method | | Advantages | Limitations |
|---|---|---|---|---|---|
| **CTR prediction** | Low-order | Logistic Regression,<br>Factorization Machines | | - Straightforward Training Process<br>- High Interpretability | - Limited to Low-order Interactions<br>- Struggles with Complex Interactions |
| | High-order | Deep Models,<br>Wide Deep,<br>DeepFM | | - High-Order Feature Interaction Learning<br>- Combines low-order model strengths with high-order interaction complexity<br>- Modular and Adaptable Design<br>- Current standard for state-of-the-art results | - Reduced Interpretability<br>- Increased Computational Demand<br>- Performance may degrade with the expansion of the model's parameter space |
| **RTB Strategies** | Static | Linear<br>&<br>Non-Linear models | | - Simplicity in Design<br>- Ease of Implementation<br>- Deployment Efficiency | - Market Dynamics Ignored |
| | Dynamic | Recurrent models | | - Demonstrates improved prediction accuracy over static methods<br>- Adaptability to Market Dynamics<br>- Efficiently models time-dependent patterns<br>- Deployment Flexibility | - Reduced Robustness<br>- Architecture restricts the seamless integration of advanced behaviors such as multi-stage planning and risk management<br>- Fails to incorporate competitive dynamics |
| | | RL based | Single Agent | - The inherent flexibility of RL frameworks fosters innovation<br>- Budget and ROI Consideration<br>- Enhanced Risk Management | - Primarily aims at optimizing outcomes for individual advertisers<br>- Overlooks Multi-Bidder Dynamics<br>- Struggles with Complex Auction Environments |
| | | | Multi Agent | - Balances competitive and cooperative agent interactions<br>- Agents continuously adapt to market dynamics | - Implementation Complexity<br>- Dynamic environments and other agents' strategies complicate learning |
| **Advertising Strategy & Bundle Recommendation** | Combinatorial Explosion | Integer Programming<br>&<br>Constraint Solvers | | - Optimized for Quantifiable Goals<br>- Precision in Constraints Handling | - Scalability Challenges<br>- The reliance on clearly defined constraints may not accommodate more nuanced or qualitative aspects of advertising strategy |
| | | Association Rule Mining | | - Discovery of Frequent Item Combinations<br>- Reveals item associations and unexpected insights without bias | - Significance Assessment Challenges<br>- Overproduction of Associations |
| | | Tensor Factorization Methods | | - High-order Interaction Modeling<br>- Efficiency with Sparse Data | - Scalability Concerns with Interaction Complexity<br>- Increase in number of features can trigger a combinatorial explosion |
| | | Traditional Framework<br>(Utility estimation &<br>Ranking) | | - Offers a straightforward and easily understandable approach<br>- High Sensitivity to Feature Variations<br>- Optimal Accuracy Within Scope | - Vulnerability to Combinatorial Explosion<br>- Cold Start Problem |
| | | Graph-Based Approaches | | - Intuitive Modeling of Relationships<br>- Complex Interaction Insights | - High Computational Demand<br>- Requires a substantial volume of data to train effectively |
| | Combinatorial Risk | LLM Based Methods | | - Offers unparalleled flexibility in generating interactive and adaptive recommendations<br>- Interactive Generation | - No inherent guarantee that all outputs will be relevant or practical<br>- Cost-intensive<br>- Sequential Nature May Limit Optimization (sub-optimal) |
| | | Genetic Algorithms | | - Excellently suited for navigating through extensive solution spaces<br>- Optimal for Multi-objective Complexity | - Premature Convergence Risk<br>- Intensive Parameter Tuning<br>- Substantial Computational Demand |
| | Combinatorial Safe | Multi-Armed Bandits | | - Balanced Exploration and Exploitation<br>- Adaptive Recommendation System | - Independent Feature Sampling<br>- Lacks Interaction Consideration |
| | | Autoregressive Methods<br>(Sequence Modeling) | | - Mitigates the risk of combinatorial explosion by generating recommendations one item at a time<br>- Incorporates Feature Interactions | - Sub-Optimality for Order-Agnostic Data<br>- Dependency on Item Order<br>- Challenges in Real-Time Efficiency |
| | | Multi-Feature Decoder Methods<br>(Attention Based) | | - Parallel Item Generation<br>- Captures global item dependencies without order reliance<br>- Scalability for Large Bundles | - Interpretability Challenges<br>- Mode Collapse Risk<br>- Complex Training Process |

Table 2.2: References of State of the Art methods in Machine Learning Applied to Digital Advertising

| Task | Category | Method | | References |
|---|---|---|---|---|
| **CTR prediction** | Low-order | Logistic Regression, Factorization Machines | | [Blondel et al., 2016, Juan et al., 2016, He and Chua, 2017, Xiao et al., 2017, Pan et al., 2018] |
| | High-order | Deep Models, Wide Deep, DeepFM | | [Guo et al., 2017, Yu et al., 2020, Wang et al., 2017b, Lian et al., 2018, Cheng et al., 2016, Song et al., 2019, Huang et al., 2019, Liu et al., 2020b, Liu et al., 2020a, Wang et al., 2021b, Bian et al., 2022, Liu et al., 2020a, Cheng et al., 2020, Wang et al., 2021a, Zhang et al., 2023b, Wang et al., 2022a, Zhang and Zhang, 2023, Wang et al., 2023, Ren et al., 2019, Ghosh et al., 2019, Xi et al., 2021, Li et al., 2022] |
| **RTB Strategies** | Static | Linear Non-Linear models | | [Perlich et al., 2012, Chen et al., 2011, Liu et al., 2017, Yang et al., 2019, Zhang et al., 2014] |
| | Dynamic | Recurrent models | | [Grislain et al., 2019] |
| | | RL based | Single Agent | [Cai et al., 2017, Liu et al., 2020c, Wang et al., 2022b, Shih et al., 2023, Du et al., 2017, Wu et al., 2018a, Wang et al., 2017c, Zhou et al., 2021, Zhang et al., 2021b, Fan and Delage, 2022, Jiang et al., 2023] |
| | | | Multi Agent | [Zhou et al., 2022, Jin et al., 2018, Wen et al., 2022, Guan et al., 2021, Zhao et al., 2018] |
| **Advertising Strategy & Bundle Recommendation** | Combinatorial Explosion | Integer Programming & Constraint Solvers | | [Marchetti-Spaccamela and Vercellis, 1995, Xie et al., 2010, Zhu et al., 2014] |
| | | Association Rule Mining | | [Fang et al., 2018, Beheshtian-Ardakani et al., 2018] |
| | | Tensor Factorization Methods | | [Wermser et al., 2011b, Chen et al., 2017, Hong and Jung, 2018, Liu et al., 2014, Pathak et al., 2017, Cao et al., 2017] |
| | | Traditional Framework (Utility estimation Ranking) | | [Wermser et al., 2011a, Covington et al., 2016b, Frolov and Oseledets, 2017, Xue et al., 2017, He et al., 2017] |
| | | Graph-Based Approaches | | [Ying et al., 2018, Gong et al., 2019, Wang et al., 2019, Zhang et al., 2022, Li et al., 2023, Ma et al., 2024, Liu et al., 2023b, Wei et al., 2023, Chang et al., 2020, Chang et al., 2023, Vijaikumar et al., 2020, Deng et al., 2020] |
| | Combinatorial Risk | LLM Based Methods | | [Zhu et al., 2023, Harte et al., 2023, Zhang et al., 2023a, Dai et al., 2023, Sanner et al., 2023, Bao et al., 2023, Yin et al., 2023, Chen, 2023] |
| | | Genetic Algorithms | | [Miralles-Pechuán et al., 2018, Ying and Zhizhong, 2009] |
| | Combinatorial Safe | Multi-Armed Bandits | | [Li et al., 2010, Qin et al., 2014, Ban et al., 2021, Deng et al., 2021] |
| | | Autoregressive Methods (Sequence Modeling) | | [Beutel et al., 2018, Hu and He, 2019, Sun et al., 2019, Katz et al., 2022, Bai et al., 2019] |
| | | Multi-Feature Decoder Methods (Attention Based) | | [Chen et al., 2019b, Arroyo et al., 2021, Chen et al., 2019a, He et al., 2019, Sun et al., 2019, Arroyo et al., 2021, He et al., 2022, Brosh et al., 2022, Wei et al., 2022a] |

Table 2.3: Our positionning in comparison to current methods

| Category | Method | Combinatorial Safe | Order Agnostic | Data Fidelity | Feature Interaction Aware | Utility Aware | Inference Exploration Mode | Inference Suggestion Aware |
|---|---|---|---|---|---|---|---|---|
| State of The Art Methods | Integer Programming Constraint Solvers | No | Yes | No | Yes | Yes | No | No |
| | Association Rule Mining | No | Yes | Some | Yes | No | No | No |
| | Multi-Armed Bandits | Yes | Yes | Some | No | Yes | No | No |
| | Autoregressive Methods | Yes | No | Some | Yes | No | No | Yes |
| | Multi-Feature Decoder Methods | Yes | Yes | Some | Yes | No | No | No |
| | LLM Based Methods | Some | No | No | Yes | No | No | Yes |
| | Tensor Factorization Methods | No | Yes | No | Yes | Some | No | Yes |
| | Traditional Framework (Estimation + Ranking) | No | Yes | No | Yes | Yes | No | Yes |
| | Graph-Based Approaches | No | Yes | Some | Yes | Some | Some | Some |
| | Genetic Algorithms | Some | Yes | No | Yes | Yes | No | No |
| **Contribution 1** | ASGAR (Base) | Yes | Yes | Yes | Yes | Yes | No | No |
| **Contribution 2** | ASGAR (Quantized) | Yes | Yes | Yes | Yes | Yes | Yes | No |
| **Contribution 3** | ASGAR (Token-Driven) | Yes | Yes | Yes | Yes | Yes | Yes | Yes |

# Chapter 3

# Data and Challenges

**Contents**

## 3.1 Introduction to the Datasets

In the realm of digital advertising, the application of machine learning techniques has become increasingly paramount. These methods rely heavily on large and diverse datasets to train algorithms capable of predicting user behavior, optimizing ad placements, and maximizing return on investment. This chapter presents the datasets employed in this thesis, underscoring the technical hurdles encountered in data processing and analysis.

### 3.1.1 Overview of the Datasets Used

Various datasets are available publicly for digital advertising related tasks. Criteo[1] provides several datasets, with the most notable being the "Kaggle Display Advertising dataset"[2] and the larger, more recent dataset "Criteo 1TB click logs"[3]. Additionally, datasets from Avazu[4], Avito[5], Outbrain[6] and TalkingData[7] have been made available through Kaggle competitions. These datasets primarily support Click-Through Rate (CTR) prediction tasks, featuring rows of ad impressions logs with a single target variable *"click"* that indicates whether an ad was clicked or not.

The iPinYou[8] dataset, published in [Liao et al., 2014] for a competition, stands out among other datasets by including standard RTB information such as bid prices, winning prices, and user information. This detail makes it particularly valuable for our objectives as it provides a continuous target variable—the winning prices—rather than just a binary

[1] `https://ailab.criteo.com/ressources/`

[2] `https://www.kaggle.com/c/criteo-display-ad-challenge`

[3] `https://ailab.criteo.com/download-criteo-1tb-click-logs-dataset/`

[4] `https://www.kaggle.com/c/avazu-ctr-prediction`

[5] `https://www.kaggle.com/c/avito-context-ad-clicks`

[6] `https://www.kaggle.com/competitions/outbrain-click-prediction`

[7] `https://www.kaggle.com/competitions/talkingdata-adtracking-fraud-detection`

[8] `https://contest.ipinyou.com/`

click variable, aligning more closely with our analytical needs (see Section 3.1.2).

While research on Click-Through Rate prediction mostly focus on Criteo's Kaggle Dataset to evaluate their results. Bidding price forecasting methods use the iPinYou dataset (alongside private proprietary datasets), we cite some of the prominent methods [Lyu et al., 2022, Shih et al., 2023, Miralles-Pechuán et al., 2023, Ren et al., 2019, Ghosh et al., 2019, Wu et al., 2015, Lin et al., 2020a, Liu et al., 2020c, Shih et al., 2020, Li et al., 2022, Fan and Delage, 2022]

Among the publicly available datasets related to digital advertising, we chose to use the iPinyou dataset [Liao et al., 2014] for its substantial size and the inclusion of cost information alongside click data. We also run our experiments on the company's private datasets.

The iPinYou dataset stands out as a cornerstone for research in real-time bidding (RTB) and programmatic advertising. Originating from one of China's largest demand-side platforms, the iPinYou dataset encompasses a comprehensive collection of data points critical for understanding and optimizing digital advertising campaigns. The dataset provides information on each ad impression, such as the context in which the ad was displayed, the type of ad, the site on which it was shown, and other relevant metadata that can be used to analyze ad performance across different contexts. Beyond simple click-through data, the dataset also includes conversion information where available, which is crucial for understanding not just which ads were effective in garnering clicks, but which actually drove meaningful engagement that led to conversions.

The dataset's granularity not only enables the detailed analysis of user behavior and ad performance but also facilitates the development and validation of sophisticated machine learning models. Figure 3.1 shows the available features in the dataset.

| Col # | Description | Example |
|---|---|---|
| *1 | Bid ID | 015300008...3f5a4f5121 |
| 2 | Timestamp | 20130218001203638 |
| †3 | Log type | 1 |
| *4 | iPinYou ID | 35605620124122340227135 |
| 5 | User-Agent | Mozilla/5.0 (compatible; \ MSIE 9.0; Windows NT \ 6.1; WOW64; Trident/5.0) |
| *6 | IP | 118.81.189.* |
| 7 | Region | 15 |
| 8 | City | 16 |
| *9 | Ad exchange | 2 |
| *10 | Domain | e80f4ec7...c01cd1a049 |
| *11 | URL | hz55b00000...3d6f275121 |
| 12 | Anonymous URL ID | Null |
| 13 | Ad slot ID | 2147689_8764813 |
| 14 | Ad slot width | 300 |
| 15 | Ad slot height | 250 |
| 16 | Ad slot visibility | SecondView |
| 17 | Ad slot format | Fixed |
| *18 | Ad slot floor price | 0 |
| 19 | Creative ID | e39e178ffd...1ee56bcd |
| *20 | Bidding price | 753 |
| *†21 | Paying price | 15 |
| *†22 | Key page URL | a8be178ffd...1ee56bcd |
| *23 | Advertiser ID | 2345 |
| *24 | User Tags | 123,5678,3456 |

Figure 3.1: Taken from [Liao et al., 2014]: Available columns in the log files.

The dataset's wide range of features and diverse campaigns result in varying performance across different features. This variability is evident in Figure 3.2, where we observe distinct fluctuations in Click-Through Rate (CTR) for different advertisers, attributed to the influence of various features.

The iPinYou dataset contains around 15M examples with 24 features in total. As shown in figure 3.3, we use 10 features in total: 8 strategy features (region, os, browser, gender, slotvisibility, slotformat, slotwidth, slotheight), 2 context features (advertiser-category, adexchange), and "payprice" as the utility score. This selection results in more than 678M theoretically possible combinations. The dataset is diverse with a variety of advertisers from different industries, which leads to the training dataset containing over 155K unique strategies when we count the different combinations present in the dataset if we group by the 10 features.

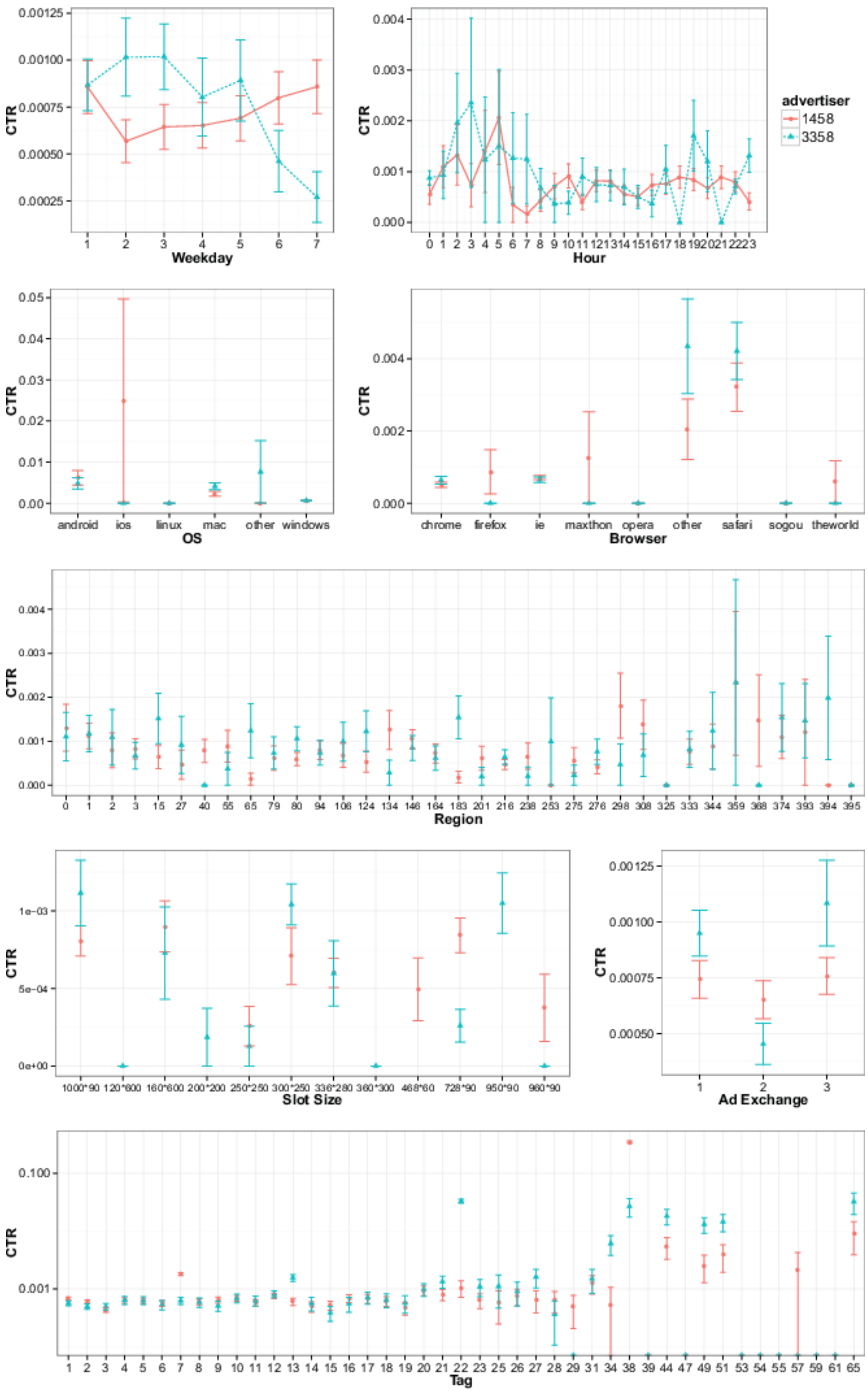



Figure 3.2: Taken from [Liao et al., 2014]: CTR distribution against different features for advertiser 1458 and 3358.

| advertiser_category_context | adexchange_context | region | os | browser | gender | slotvisibility | slotformat | slotwidth | slotheight | score |
|---|---|---|---|---|---|---|---|---|---|---|
| Footwear | 4.0 | 374 | windows | firefox | Female | FirstView | Na | 300 | 250 | 120.000000 |
| Software | 2.0 | 146 | windows | theworld | Male | 1 | 0 | 728 | 90 | 81.625000 |
| Telecom | 1.0 | 65 | mac | other | Unknown | FirstView | Na | 300 | 250 | 80.000000 |
| Footwear | 3.0 | 0 | windows | chrome | Unknown | Na | Na | 300 | 250 | 127.613537 |
| Footwear | 4.0 | 216 | windows | safari | Female | SecondView | Na | 300 | 250 | 36.000000 |

Context Features Strategy Features Utility

Figure 3.3: Our chosen context and strategy features and performance score column.

The Programmatic Company's private dataset contains 19M examples in total. We use only six strategy features for the current deployment of the model for business needs, two context features, and "Cost per Click" as the utility score. Resulting in only 785K possible combinations, of which only 8K are present in the training dataset due to the relatively low diversity of our advertisers' profiles. In contrast to the readily accessible iPinYou dataset, acquiring and processing the company's private datasets demanded considerable effort in the initial phases of this thesis. The technical challenges encountered during this period will be discussed further.

### 3.1.2 Rationale for Dataset Selection

The choice of the iPinYou dataset for our research was driven by specific critical factors. Primarily, our study required a dataset that includes information on ad pricing. Most publicly available digital advertising datasets feature merely a binary click target variable. While other datasets related to recommendation tasks do offer a continuous target variable, such as movie ratings in MovieLens[9] Datasets, they typically only include two or three features. While we could have opted to use click-related datasets and augment them with a calculated Click-Through Rate (CTR) column, we chose to avoid the potential complications and noise that come with such data at the onset of this thesis. Instead, we selected the price variable, which despite still being noisy is more stable and

[9] `https://grouplens.org/datasets/movielens/`

exhibits lower variance compared to the CTR variable, thus simplifying our tasks.

The iPinYou dataset is not only similar to the company's private dataset but also contains a large volume of data points, providing an excellent foundation for applying and evaluating our research contributions. The variety within the dataset, including records from numerous advertising campaigns, diverse user demographics, and various contexts, creates a realistic and complex testing ground. This diversity is ideal for assessing the robustness and scalability of the methods we propose.

Furthermore, its frequent use in previous research work ensures that our results can be replicated and verified by other researchers. This reproducibility also allows for meaningful comparisons with existing and future studies, helping to contextualize our findings within the broader research landscape.

Lastly, the relevance of the iPinYou dataset to current industry challenges, such as optimizing real-time bidding (RTB) mechanisms, refining user targeting techniques, and boosting the effectiveness of advertisements, perfectly aligns with the goals of this thesis.

## 3.2 Data Collection Process

### 3.2.1 Sources of Data: Primary vs. Secondary

In The Programmatic Company, we use several advertising platforms. Often, an advertising campaign is distributed across these platforms to broaden audience reach and create multiple points of engagement to enhance awareness.

This distribution across various platforms leads to a significant challenge: the datasets we receive are formatted differently. Additionally, while some platforms may provide certain data features, others might lack them. This heterogeneity necessitates the creation of a unification pipeline to standardize the datasets. To compensate for missing or obscured features, we employ strategies to gather supplementary data either from secondary data provided by the platforms or from external sources and integrate it where necessary.

A straightforward example of utilizing a secondary data source is the inclusion of holiday dates within the dataset. By adding a "is_holiday" flag, we can enrich the dataset's temporal dimension, offering a clearer context for ad performance during holiday periods.

The overall data flow shown in Figure 3.4 illustrates how a single advertising campaign results in multiple data sources.

### 3.2.2 Techniques and Tools for Data Collection

This section outlines the methods and tools utilized for data collection, focusing on JSON, SQL, and the Dask Python library for efficient parallel data processing.

**JSON and API calls:** Multiple platforms provide data in the form of a large JSON object as a response to an API call. Taking Meta as an example, we retrieve various types of data insights in JSON format, including both detailed breakdowns and summaries of overall performance. Our approach seeks to capture the most detailed and raw data available, subsequently filling in missing features with data from the breakdowns, which include some attributes absent in the raw data. To explore potential insights from feature interactions, we introduce the extracted attributes from breakdown into duplicated raw-data rows, which does not introduce any bias nor adds predictive value. We anticipate that this approach will allow our model to discern how these introduced features interact with existing ones across various rows. This method facilitates the simulation of disaggregated data, enabling more detailed analysis. Our objective is to keep metrics averaged across features, focusing on preserving the completeness of data rather than omitting features due to the absence of certain data points.

**Dask for Parallel Data Collection:** Dask, a dynamic parallel computing library in Python, enhances our ability to manage and process large datasets effectively. It offers

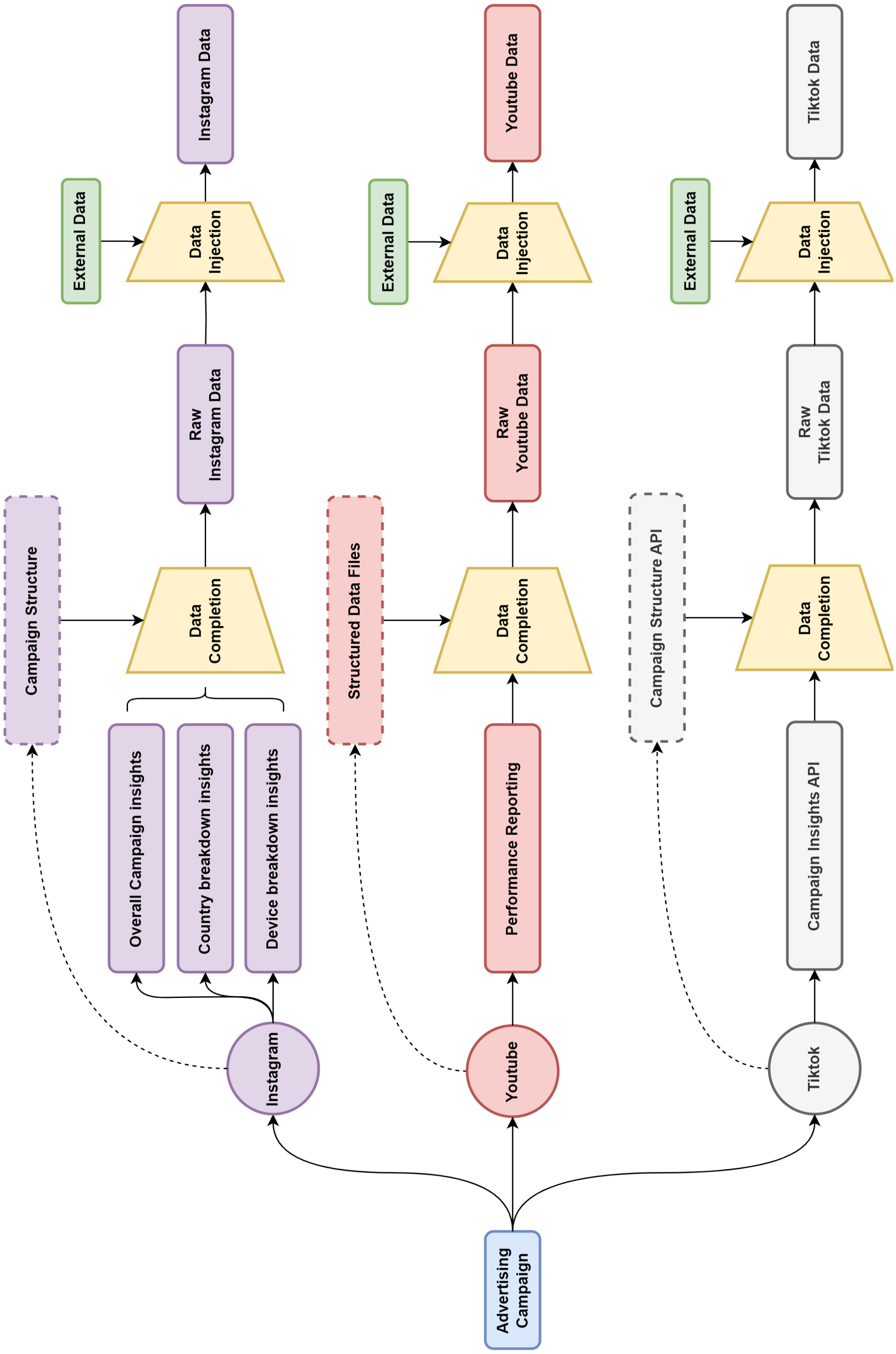



Figure 3.4: Data Flow process for extracting Data from different advertising platforms.

parallel computing solutions that integrate seamlessly with existing Python libraries, such as Pandas and NumPy. By utilizing Dask, we can perform parallel data collection and processing tasks, significantly reducing the time required to prepare datasets for analysis. This is particularly beneficial when dealing with the large and diverse datasets characteristic of digital advertising platforms.

**PostgreSQL for Data Management:** Once datasets are extracted and converted into a tabular format, they are stored in a PostgreSQL database for faster access and laying the groundwork for the upcoming step of data unification.

## 3.3 Data Cleaning and Preprocessing

### 3.3.1 Identifying and Handling Missing Values

In our data preprocessing phase, managing missing values is a critical step to ensure the integrity and utility of our datasets. Our approach to handling missing values is twofold, focusing on both individual data points and entire features that might be absent.

When encountering a missing value, our initial step is to assess whether it can be substituted with a default value that does not skew the dataset's overall statistics. For instance, an empty "Gender" field often indicates the absence of gender-specific targeting rather than an actual missing value. In such cases, we assign a "Non-Targeted" value to reflect this intention accurately. If substituting a meaningful default is not feasible, we opt for a "None" placeholder and leave the task to the modeling phase to deal with censored/sparse data.

If an entire feature is missing from the dataset, our method involves inferring the missing data from alternative sources or prior knowledge. For example, if the "Device" column is absent and we know from the campaign's setup that some specific advertising strategies (rows) exclusively targeted "Desktop" users, we confidently fill this information in the corresponding rows.

### 3.3.2 Data Transformation and Normalization

Our approach to data normalization and transformation is designed to address the challenges posed by the disparate formatting and structuring of datasets originating from various advertising platforms. This process is essential for harmonizing our data, enabling comprehensive analysis and modeling. Here's how we enhance the consistency and utility of our datasets:

**Unification of Features:** The first step in our normalization process involves standardizing the naming conventions across different datasets. For instance, a feature labeled as "Country" in one platform might be referred to as "GeoTargeting" in another. By identifying and unifying these equivalent features, we establish a common ground for further analysis. This step is crucial for facilitating cross-dataset comparisons and aggregations.

**Normalization of Attribute Values:** Beyond the unification of feature names, we delve into the normalization of attribute values themselves. This involves a significant effort to map and translate the data structures from various ad platforms into a standardized format. For example, country names or codes may need to be harmonized across datasets to ensure consistency. The complexity of this task cannot be understated, as it requires an exhaustive understanding of each platform's data schema and the creation of comprehensive mapping strategies to align the data.

**Handling Platform-Specific Features:** Despite our efforts to unify and normalize data, we acknowledge that some features are exclusive to certain platforms. To maintain the integrity and density of our dataset, we opt to exclude these platform-specific features for the time being. Our rationale is to prioritize the creation of a cohesive and robust dataset that maximizes the amount of usable data. By focusing on shared features across platforms, we aim to build a dense dataset conducive to model training. This exclusion decision was made by weighing the benefit of having a wide variety of information against the potential drawbacks of having sparse data with rare feature attributes.

**Simplifying Complex Features:** We encountered scenarios where individual features were either represented as lists of values or aggregated in a manner that prevented further disaggregation. Ideally, such complex features would be best analyzed using models designed to handle graph-like or hierarchical attributes, acknowledging the intricate relationships and dependencies between its inner values. However, given the challenges already present in the daunting task of digital advertising strategy recommendation, we opted for a pragmatic simplification approach by considering the whole list as a single value. Importantly, we made sure to reorder similar lists all in the same order to eliminate the risk of considering two lists as different while they have the same combination of values.

For cases where a feature, like "interests," contained a lengthy list of values, we made a strategic decision to exclude these instances from our analysis. This decision was informed by the recognition that such cases likely represent outliers, as it's rare for other advertising strategies to share an identical, extensive list of interests.

Despite this simplification, our approach still enables valuable insights to be drawn from the data. For instance, the "Device" feature frequently presented combined values such as "Desktop & Mobile" across different strategies. By treating these combined entries as distinct values, we could preserve meaningful information about the advertising preferences and strategies that span multiple device types.

## 3.4 Technical Challenges in Data Handling

Given the voluminous nature of both datasets, encompassing over 15 million rows of logs, the primary challenge lay in the efficient extraction and processing of this data. To address this, we leveraged parallel computing capabilities offered by Dask, which significantly sped-up the data handling process. We stored our various data files in Azure Blob Storage and then employed Dask to process these files and transform the data into a structured tabular format. Once we had a consolidated dataset, we chose to store it in an SQL database, which is well-suited for managing large volumes of tabular data without sacrificing performance.

The datasets predominantly consist of log entries, leading to a notable degree of redundancy. Specifically, identical combinations of strategy features frequently recur within the same day or even hour, each instance paired with varying performance KPIs influenced by the time of day or day of the week. To manage this data at scale and simplify the dataset, we transformed the temporal dimension into categorical features (e.g., hour of the day, day of the week) instead of maintaining each unique occurrence. This approach not only facilitates data handling and analysis but also aligns with business objectives by enabling the identification of optimal time frames for advertising strategies, thereby offering valuable insights for strategic planning.

## 3.5 Ethical and Legal Considerations

Regarding the ethical aspects of our data processing, it is important to note that our approach was largely shaped by the practices and policies of the advertising platforms from which we sourced our data. These platforms have established protocols for filtering datasets and anonymizing sensitive information to comply with privacy regulations such as the General Data Protection Regulation (GDPR).

As a result, the datasets we worked with were already processed to remove or anonymize sensitive features, ensuring that the information was GDPR compliant. This preprocessing by the platforms significantly reduced the burden on our side to implement additional ethical safeguards. However, it also underscored the importance of relying on data sources that prioritize user privacy and data protection.

## 3.6 Conclusion

In this chapter we presented an overview of the datasets that underpin this thesis. The use of the iPinYou dataset, complemented by the company's private datasets, provides a solid foundation for exploring and validating the efficacy of our contributions.

### 3.6.1 Summary of Data Handling Efforts

We presented the process of selection, collection, preprocessing, and the technical considerations that shaped our approach. Starting with raw acquisition of heterogeneous datasets from diverse advertising platforms to the unified, processed datasets ready for analysis. We explained some of our decisions about handling missing features and simplifying complex features aiming to retain as much valuable information from the data as possible while minimizing excessive filtration.

### 3.6.2 Impact on Research Outcomes

Utilizing a public dataset enables the validation and reproducibility of our methodologies by the broader research community. The extensive variety and depth of the iPinYou dataset facilitate a thorough examination of our methods' robustness more precisely than on the company's private dataset, which could reflect a bias toward our existing clientele.

# Chapter 4

# Advertising Strategy Generation

## Contextual Advertising Strategy Generation via Attention and Interaction Guidance

### Contents

The content of this chapter elaborates on the work presented in [Benamara and Viennet, 2023]. In this work, we proposed a novel framework and generative model which leverages the attention mechanism and a guiding network to contextually generate optimal advertising strategies while avoiding combinatorial explosion. We evaluate our results on a public dataset and, in the absence of directly comparable methods, compare against the state-of-the-art methods that comply with the majority of the task's constraints.

This chapter is organized as follows: We start with an introduction in Section 4.1. In Section 4.2 we discuss related work. In Section 4.3 we explain our approach. In Section 4.4 we discuss our experimental results and compare to state-of-the-art methods, and finally we provide a conclusion in Section 4.5.

## 4.1 Introduction

As previously outlined in Chapter 1, the digital advertising industry has witnessed exponential growth, necessitating the development of more sophisticated and efficient strategy design methodologies. While the initial chapters of this thesis have set the stage by delving into the related works in digital advertising and their relationship to this thesis, detailing the challenges inherent in strategy design, and defining the constraints for a successful approach, this chapter focuses on a practical application of these concepts. It showcases our novel method for contextually generating advertising strategies, leveraging a transformer-based model that incorporates attention and interaction guidance mechanisms.

Digital advertising works by displaying ads on websites, social media platforms, and other digital channels. When a business wishes to advertise their product, they have to set up an advertising campaign. They define the goal of the campaign (e.g. increase sales, brand awareness, etc.). They identify the audiences to target by specifying multiple criteria (e.g. country, age, interests, etc.). Audiences are chosen on the basis of how likely they are to be interested in the product. Finally, they choose one or many diffusion platforms (e.g. Instagram, Google Search, websites, etc.) that will display the ads. Diffusion platforms are where advertisers choose more precisely who, when, and how to target by specifying diffusion dates for the ad campaign or more technical features, such as real-time bidding parameters. A combination of all these criteria/attributes (e.g. France, Women, 18-25 years old, Interested in gaming, etc.) is called an advertising strategy. An advertising campaign consists of multiple advertising strategies, all aiming to achieve the same overall goal.

Despite the importance of digital advertising, the design process of advertising strategies still relies heavily on human expertise. It involves the manual selection of strategy features based on prior experience, intuition, and market research with some basic statistical methods. This approach is inefficient, as it requires significant time and resources, as well as the challenge of combinatorial exploration of the feature space, leading to suboptimal campaign performance.

The problem of contextual advertising strategy generation/recommendation can be described as follows: given a context (e.g. advertiser industry, diffusion platform, etc.) and a campaign goal KPI (e.g. Cost Per Click), generate a set of advertising strategies that perform the best in terms of the goal KPI. This problem poses many challenges, both industrial and scientific, which impose a few constraints.

In Section 1.1, we identified five key constraints for an effective approach. In this chapter, our suggested method addresses the four main constraints, notably:

- Be **non-combinatorial**: explore the strategy feature space without literally exploring all possible combinations.
- Be **order-agnostic**: feature order should be disregarded, both as input and during generation.
- Offer **parameterizable exploration and exploitation**: to control how much we want to replicate known strategies and how much we want to discover new strategies.
- Model **feature interactions**: because a single feature swap can affect the strategy performance.

In this chapter, we propose a novel method and a framework for contextually generating advertising strategies. We propose a neural network model for Contextual Advertising Strategy Generation via Attention and InteRaction Guidance (**ASGAR**). Our main contributions are:

- We propose a framework composed of two interacting submodules: a strategy performance estimation module that can be any state-of-the-art regression model and a strategy generation module which leverages the attention mechanism through transformers.

- We propose a novel loss function that uses the estimator module for exploration guidance during the training of the generation module through smooth contrastive learning while allowing for controllable exploration and diverse generation.
- To the best of our knowledge, this is the first approach that satisfies all the constraints of non-combinatorial exploration, order-agnostic generation, controllable exploration, and feature interaction modeling.

We evaluate our method on a public dataset iPinYou [Liao et al., 2014] as well as the company's private dataset (see Section 3.1.1). In the absence of directly comparable methods that satisfy all the constraints, we benchmarked our results against prominent methods from other fields, adapted for this specific task.

## 4.2 Related Work

Given the comprehensive exploration of related works and the state-of-the-art in digital advertising strategy design discussed in Chapter 2, this chapter will streamline the paper's original related work section to avoid redundancy, focusing instead on highlighting how this particular study advances our overall research objectives.

**Traditional methods of recommender systems** have been widely used in many candidate generation problems due to their capacity to estimate the utility of any given candidate and capture some level of feature interactions. From tensor factorization techniques [Wermser et al., 2011a, Frolov and Oseledets, 2017, Hong and Jung, 2018], to deep neural networks [Katz et al., 2022, Brosh et al., 2022, Guo et al., 2021, Deng et al., 2020], these methods have proven efficient and reliable. However, during the ranking (generation) phase, they suffer from combinatorial complexity as there is a need to calculate the utility for all the candidates to rank them, which proves impossible when the number of possible candidates is too high to explore in its entirety as in the advertising strategy feature space. Some methods used prefiltering methods to reduce the number of candidates before utility estimation [Covington et al., 2016a, Pathak et al., 2017]. However, it either requires prior knowledge for filtering or the capacity to assign a utility score to subsets of features.

**Evolutionary approaches**, such as Genetic Algorithms and Evolutionary Strategies, have been used due to their ability to handle combinatorial optimization problems. However, these methods suffer from slow convergence rates, the risk of premature convergence to local optima, and the inability to control exploration and exploitation rates in an efficient manner.

**Contextual Multi-Armed Bandits** (MAB) methods have been widely used in digital advertising problems [Avadhanula et al., 2021, Nuara et al., 2022, Ban et al., 2021], especially in real-time bidding problems, due to their ability to alleviate the combinatorial exploration problem with their adaptability to high-dimensionality features and exploration-exploitation trade-offs. They also model probability distributions for each type of strategy feature, therefore, allowing sampling techniques to generate diverse strategies while still optimizing the same goal. Reference [Lowe et al., 2017] trains multiple bandits with indirect coordination to optimize the same global utility score; [Ban et al., 2021] improved on that by jointly learning multiple bandits' reward functions to coordinate their arms in a more direct way. Unfortunately, MABs fail to capture feature interactions (at least not in an efficient way) because of the independent arms and agents and cannot truly coordinate all the bandits, which leads to sub-optimal results.

Recently, **Attention mechanisms** [Vaswani et al., 2017] have been widely used to enhance the performance of various methods, including traditional recommender systems and autoregressive generative approaches [Hu et al., 2022, Arroyo et al., 2021, Devlin et al., 2019]. These mechanisms enable the capture of complex feature interactions and dependencies, leading to improved performance in almost every state-of-the-art approach.

**Variational Auto-Encoder methods** have regained popularity again thanks to the improvements brought about by the attention mechanisms. Their advantages for advertising strategy generation are single-shot candidate generation and feature interaction modeling. Reference [Arroyo et al., 2021] is able to model document layout design rules as a distribution, rather than using a set of predetermined heuristics, increasing the diversity of the generated layouts. However, their method still lacks enough control over the exploration rate, as it generates samples that are highly similar to the training data.

**Autoregressive generative methods** have shown impressive results in text generation. References [Devlin et al., 2019,Köpf et al., 2023] are state-of-the-art large language models that leverage the attention mechanism to learn complex relationships between the different works in sentences, then generate the next token in an autoregressive manner. However, when applied to the advertising strategy problem the autoregressive nature these methods yields sub-optimal results by design as they assume an ordering between strategy features.

**Personalized Bundle Recommendation** approaches can be analogous to contextual advertising strategy generation [Brosh et al., 2022,Deng et al., 2021]. They typically use graph neural networks to model feature interactions and generate bundles of items that maximize a given goal (in our case, they generate bundles of strategy features). Unfortunately, graph neural network approaches are often too complex to train successfully for high-dimensional data such as in digital advertising. Other approaches of Personalized Bundle Recommendation fall back to the other cited methods.

## 4.3 Methodology

This section illustrates the framework and contextual advertising strategy generation method **ASGAR**. The framework consists of two submodules:

- **Performance Estimator:** a module that given a context and a strategy, estimates the utility of the strategy relative to the context. The utility can be a KPI of business advertising performance or a formula that mixes multiple KPIs.
- **Contextual Strategy Generator:** a module that, given a context only, generates an advertising strategy.

The performance estimator models feature interactions to estimate the utility score. It can be any state-of-the-art regression model, but preferably with great results, as the estimator acts as an oracle during the training of the strategy generator to guide it during its exploration of the feature space.

### 4.3.1 Definitions

Consider an advertising dataset $D$ that consists of $N$ instances from $(C, S, u)$. Where $C$ is the set of all possible contexts, $S$ represents the set of strategies, and $u \in \mathbb{R}$ is the utility score representing the performance of a given strategy $s \in S$ in a given context $c \in C$.

**Context features** define a context $c \in C$ such as there are $n_C$ features. Therefore $|c| = n_C$, and $c = \{c_0, c_1, .., c_{n_C}\}$ where $c_i$ is a categorical feature. Context features typically describe the advertised product type, the advertiser's industry or some features that are imposed conditions (e.g. platform of diffusion).

**Strategy features** define a strategy $s \in S$ such as there are $n_S$ features. Therefore $|s| = n_S$, and $s = \{s_0, s_1, .., s_{n_S}\}$ where $s_i$ is a categorical feature. Strategy features define the required configuration parameters to setup an advertisement for display (e.g. country, age range, interests, budget pacing type, etc.).

**Performance score** or **utility score** $u$ is the KPI value resulting from the strategy $s$ being displayed in a context $c$. It is the KPI that we wish to optimize (e.g. cost per click).

Let $E : C \times S \to \mathbb{R}$ be the Performance Estimator model. Which takes as input a context $c$ and a strategy $s$ and outputs the estimated utility $u^E \in \mathbb{R}$ if the strategy $s$ is displayed in context $c$.

Let $G_\theta : C \to S^k$ be the Strategy Generator model. Which takes as input a context $c$ and outputs a set of $k$ strategies $\{s \in S\}_k$. The parameters $\theta$ dictate the generative behavior of the model (e.g. balance estimator guidance, balance exploration / exploitation).

The framework's goal is to generate a set of strategies that have the best utility for a given context, ie:

$$\forall s^G \in G_\theta(c),\ E\left(c, s^G\right) \in [\max(k,\ U_c),\ \max(1,\ U_c)] \tag{4.1}$$

Such that $s^G$ is a generated strategy, $U_c = \{E(c, s),\ \forall s \in S\}$ is the set of all possible utility scores for a given context $c$, $\max(k,\ U_c)$ is the $k$ $^{th}$ highest utility score in the set of all possible utility scores for the given context $c$. The goal is to setup an advertising campaign consisting of the best strategies for the given context.

### 4.3.2 Performance Estimator

The goal of the Performance Estimator is to accurately estimate the performance score of a context-strategy couple by modeling feature interactions. It plays the important role of being the Oracle that guides the Strategy Generator during its exploration of combinatorial space. The idea being that if the Strategy Generator stumbles upon a strategy that has an interesting estimated performance score, it weighs the strategy Generator's loss function such that it dictates if it needs to explore farther in the strategy space.

Any state-of-the-art regression model can be used as a Performance Estimator as long as it provides low margins of error in inference with great generalization power.

We employed a regression model adapted from state of the art Click-Through Rate forecasting models (see Section 2.2.1) trained using Mean Squared Error (MSE) as the loss function. The model was trained to optimize the following loss function:

$$\min \ \mathcal{L} = \sum_{i=1}^{N} (u_i - E_\theta(c_i, s_i))^2$$

Here, $E_\theta$ represents the Performance Estimator, which outputs predictions for the utility scores based on the given context and strategy. The objective is to minimize the squared differences between the predicted utility scores and the actual utility scores across all data points, thereby enhancing the accuracy of the predictions for the given context-strategy combinations.

### 4.3.3 Contextual Strategy Generator

The goal of the Contextual Strategy Generator is that, given a context, it generates a set of strategies that perform the best in the given context in terms of utility score.

The Generator must also satisfy all the imposed constraints: Non-combinatorial, non-autoregressive, variational, and offer controllable exploration/exploitation parameters.

Our proposed model ASGAR is adapted from the base architecture of [Wei et al., 2022a] to the contextual advertising strategy generation problem. We differ on the loss formula and the training process. The architecture we used is described in Fig. 4.1, it is what satisfies the non-combinatorial and non-autoregressive constraints. For a given row of the dataset, only the context features values are used in the input. The strategy features vocabularies are always fed in their entirety to the model. Therefore, the model never sees the strategy features values directly. Feature-type specific embeddings are learned to serve as triggers in the decoder part of the architecture. Self-attention here is used to model dependencies between a context and all possible strategy features values, while cross-attention allows the decoder to attend to the output of the encoder, considering the provided triggers.

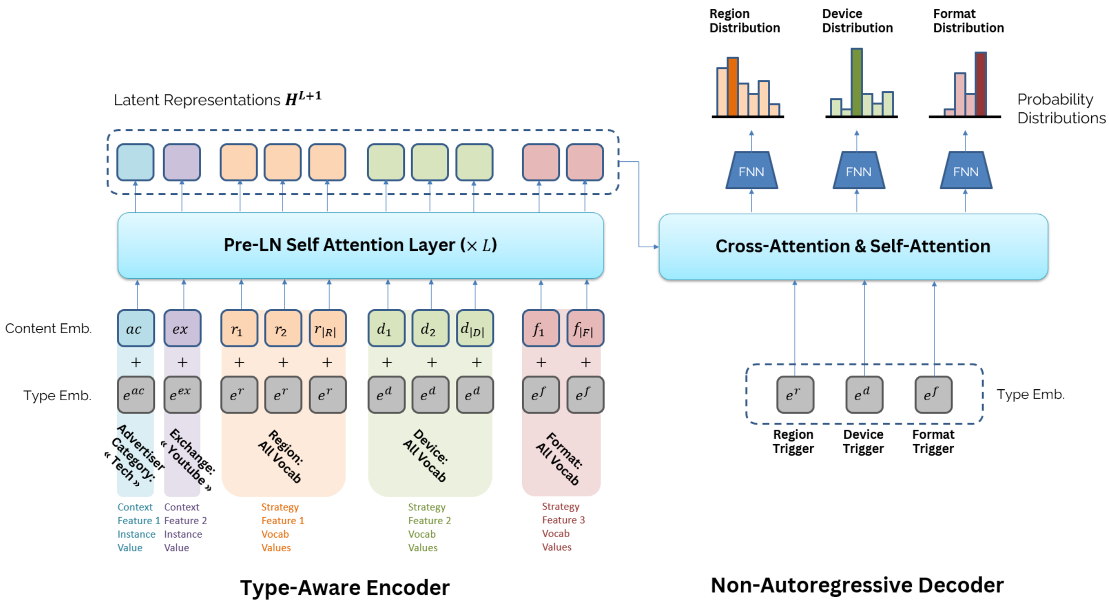



Figure 4.1: Overview of our Contextual Advertising Strategy Generator ASGAR taking as input a context (blue and purple) along with all the possible strategy features values (orange, green, and red) to use the attention mechanism and output distributions of strategy features.

The attention mechanism is highly important in this architecture as it is what allows the model to capture the relationship between contexts and strategies and improves on inner feature interactions. It lets the model focus on specific strategy features and their values that are most promising for a given context.

Our proposed loss formula for training the Generator is pivotal, as it utilizes the Performance Estimator as a guiding Oracle and facilitates controllable exploration and exploitation through hyper-parameters $\theta$. This formula also indirectly incorporates the feature interactions learned by the Performance Estimator, enhancing the effectiveness of our model.

Training is carried out in a smooth contrastive learning process. The model can be trained using mini-batches, but for clarity we illustrate the iterative process for a single example $(c, s, u)$ from the dataset $D$:

1. The Strategy Generator $G_\theta$ takes $c$ as an input and outputs a generated strategy $s^G$

2. The Performance Estimator $E$ then takes $(c, s^G)$ as an input and outputs the estimated generated strategy utility score $u^G$

3. The proposed loss formula takes into account $(c, s, u, s^G, u^{G)})$: the context, the observed strategy in the dataset, the observed utility score in the dataset, the generated strategy, the estimated utility for the generated strategy

4. The loss function behaves in a way to either encourage or discourage the generation of certain strategies, which during the backward pass for the Strategy Generator Network either pushes $s^G$ to be more similar to $s$, or changes its output to explore the feature space and find better performing strategies.

To diversify model outputs for the same input, we incorporate a variational component, similar to the approach described in [Fang et al., 2021]. This method maps inputs into a latent space of parameters of a normal distribution (mean and variance). During inference, we sample directly from this latent space using the predetermined mean and variance to generate varied outputs. However, as detailed in Section 4.3.4, our loss formula aims to achieve dual objectives. It either reconstructs a strategy or generates a new promising one, making this variational approach susceptible to mode collapse. To counteract this, we also employ dropout as a means to mimic the training of multiple models, each converging differently and thus producing distinct results.

### 4.3.4 Loss Formula

In order to clearly describe how the loss formula works, we need to define a few concepts first.

Let $R : S \times S \to \mathbb{R}$ be a dissimilarity function that scores how different two given strategies are. Lower values mean closer strategies. In our case we used the averaged cross entropy over all strategy features:

$$R(s_1, s_2) = \frac{1}{n_S} \sum_{i=1}^{n_S} H(s_{1,i}, s_{2,i}) \tag{4.2}$$

Let $P : C \times \mathbb{R} \to [0, 1]$ be the positiveness function. We define the concept of "positiveness of a strategy" as a value between 0 and 1 that scores how good a utility score is relative to a context. The same utility score can be considered as a good performance in context A, but it can still be considered a bad performance in context B. Therefore, for a tuple $(c, s, u)$, $P\left(c, u\right)$ indicates how good is the strategy $s$ in context $c$ and $P\left(c, E\left(c, s^G\right)\right)$ indicates how good is a generated strategy $s^G$ relative to a context $c$. In our case we define the positiveness function as follows:

$$\overline{u_c} = \frac{\sum_{1 \leqslant i \leqslant N,\ c_i\ =\ c}\ u_i}{\sum_{1 \leqslant i \leqslant N,\ c_i\ =\ c}\ 1} \tag{4.3}$$

$$g(c) = \frac{\log\left(\frac{1}{p_f} - 1\right)}{\overline{u_c} - u_f} \tag{4.4}$$

$$P(c, u) = \frac{1}{1 + e^{g(c).(\overline{u_c}\ - u)}} \tag{4.5}$$

Where $\overline{u_c}$ is the average utility score of context $c$ for all its observed strategies in the training dataset.

We define the function $g$ to create a fixed point in our sigmoid-like positiveness function $P$ and for a cleaner writing. Where $p_f$ and $u_f$ are parameters to fix a point in the positiveness curve, $p_f$ is equal to the positiveness score we wish to assign to a $u_f$ utility score. For example, if $p_f = 0.95$ and $u_f = 130$ it means that a utility score of 130 will have a positiveness score of 0.95.

Finally, we define our loss formula as a linear combination of 3 sub-functions as follows:

$$\mathcal{L} = pos.\mathcal{L}_{pos} + avg.\mathcal{L}_{avg} + neg.\mathcal{L}_{neg} \tag{4.6}$$

This linear combination acts as an interpolation between the three sub-functions since the weights are forced to satisfy $pos + avg + neg\ =\ 1$. They are used to assign a sub-function to extreme cases and interpolate between them. For example, for a very positive strategy $s$, $\mathcal{L}_{pos}$ must be the most active term in the loss formula, so we want $pos = 0.98, avg = 0.01, neg = 0.01$. In our case we define them as follows:

$$pos(c,u) = \begin{cases} 2.P(c,u) - 1 & if\ P(c,u) \geqslant 0.5 \\ 0 & else \end{cases} \tag{4.7}$$

$$neg(c,u) = \begin{cases} 1 - 2.P(c,u) & if\ P(c,u) \leqslant 0.5 \\ 0 & else \end{cases} \tag{4.8}$$

$$avg(c,u) = 1 - pos(c,u) - neg(c,u) \tag{4.9}$$

The idea for this interpolation is to activate a different loss behavior depending on the currently encountered strategy in the dataset. For example, if the strategy has a high utility score, then $\mathcal{L}_{pos}$ is active, and how this sub-function is defined is what drives the Generator to either replicate the same strategy or to be more exploratory and generate a different strategy.

The sub-functions $\mathcal{L}_{pos}, \mathcal{L}_{avg}, \mathcal{L}_{neg}$ dictate how the Generator learns when encountering a great strategy, a neutral strategy, and a bad strategy in the training dataset. In our case we wish to push the Generator towards replicating great strategies, either be close to a neutral strategy or be flexible in proposing new ones without being penalized, and avoid bad strategies. To do so, we define the sub-functions as follows:

$$\mathcal{L}_{*}\left(c, s, s^{G}, u^{G}\right) = f_{*}\left(R\left(s, s^{G}\right), P\left(c, u^{G}\right)\right) \tag{4.10}$$

$$f_{pos}(x,y) = e^{\frac{y - w_{pei}}{w_{pei}}} + w_{pdf}.x^{2}.\left(y^{2}+1\right) \tag{4.11}$$

$$f_{avg}(x,y) = e^{\frac{y-w_{aei}}{w_{aei}}} + \frac{w_{aa}}{x+1} \tag{4.12}$$

$$f_{neg}(x,y) = \frac{w_{nf}.e^{w_{ne}.y}}{x+0.1} + y.e^{w_{ne}.y} \tag{4.13}$$

There are 6 hyperparameters in total to control the learning behaviour of the Generator.

- When the current strategy is positive in (4.11) $w_{pei}$ is a hyperparameter (positive estimator influence) to control how much we trust the Estimator's guidance for exploring other strategies and $w_{pdf}$ is a hyperparameter (positive data fidelity) to control how much we want to strictly replicate the positive strategy or be flexible about exploring other strategies.
- When the current strategy is average in (4.12) $w_{aei}$ is a hyperparameter (average estimator influence) to control how much we trust the Estimator's guidance for exploring other strategies and $w_{aa}$ is a hyperparameter (average data avoidance) to control how much we want to differ from the average strategy or be okay with replicating it.
- When the current strategy is negative in (4.13) $w_{ne}$ is a hyperparameter (negative estimator influence) to control how much we trust the Estimator's guidance for exploring other strategies and $w_{nf}$ is a hyperparameter (negative data fidelity) to control how much we want to differ from the negative strategy.

Figure 4.2 shows the intuition behind the 3 sub-functions that illustrates how the gradients flow and thus dictate the model's behavior toward learning to replicate a strategy or avoid it. Figure 4.3 shows how the interpolation behaves depending on the positiveness of the current strategy, which customizes the loss formula for each strategy.

In addition to this loss formula, we add a Kullback-Leibler term for the variational component of the Generator. We use the method described in [Cipolla et al., 2018] to balance

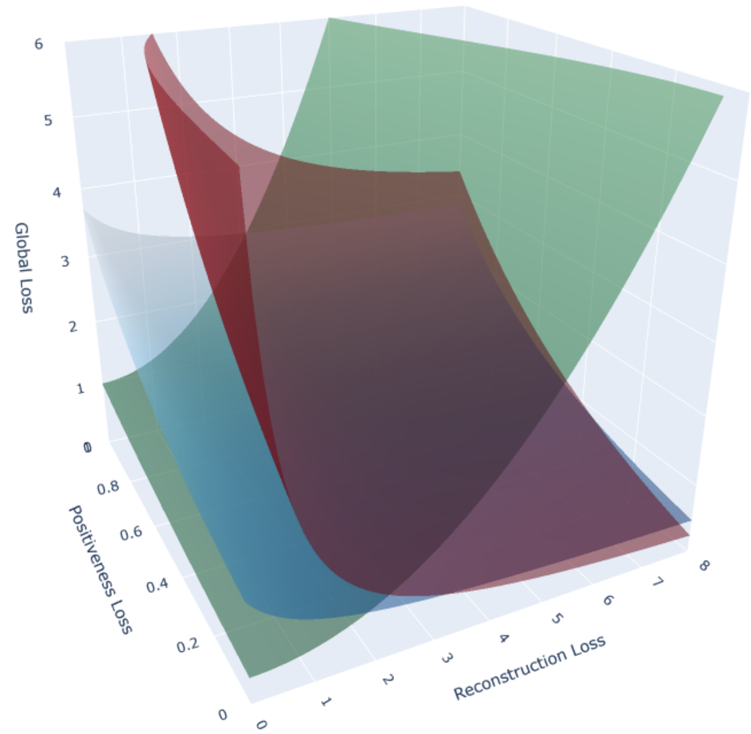



Figure 4.2: 3D curves of $f_{pos}$ (green), $f_{avg}$ (gray), and $f_{neg}$ (red) relative to the estimated utility score on the $x$ axis and dissimilarity score on the $y$ axis.

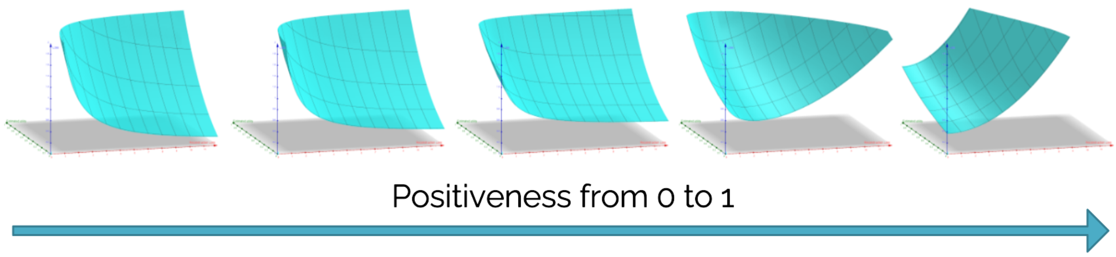



Figure 4.3: The interpolation of the 3 subfunctions leads to a gradual change in the loss formula as positiveness goes from 0 to 1.

the Kullback-Leibler term in the final loss function. This method uses homoscedastic uncertainty as a task-dependent weight, which naturally balances the contribution of each task's loss based on its uncertainty. By treating the uncertainty levels of different tasks as variables to be learned, the network can automatically adjust the importance of each task during training. This results in a more efficient learning process and potentially better performance across all tasks compared to training them individually or with fixed, manually-tuned weighting schemes. Therefore, instead of manually tuning the balancing weights, the weightings are implicitly learned using this method. This resulted in superior performance in our experiments.

## 4.4 Experimental Results and Evaluation

### 4.4.1 Datasets and Evaluation protocol

#### 4.4.1.1 Datasets

We evaluate our method on the following publicly available digital advertising dataset.

IPinYou [Liao et al., 2014] contains 15M examples with 24 features in total. We use 8 strategy features, 2 context features, and "payprice" as the utility score, resulting in more than 678M possible combinations. The dataset is diverse with a variety of advertisers from different industries, which leads to the training dataset having 155K unique strategies.

Table 4.1 shows the volume and distribution of strategy scores across various contexts in the iPinYou dataset. We observe a high level of noise represented by the extreme minimum and maximum values.

The Programmatic Company's private dataset contains 19M examples in total. We use only six strategy features for the current deployment of the model for business needs,

| advertiser_category_context | adexchange_context | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|---|
| Chinese vertical e-commerce | 1.0 | 822685.0 | 68.504595 | 56.796751 | 5.0 | 25.0 | 55.0 | 87.0 | 300.0 |
| | 2.0 | 724389.0 | 79.143634 | 69.025554 | 5.0 | 28.0 | 55.0 | 126.0 | 300.0 |
| | 3.0 | 902860.0 | 62.169752 | 29.149092 | 5.0 | 50.0 | 70.0 | 70.0 | 282.0 |
| Footwear | 1.0 | 202969.0 | 118.654785 | 83.792006 | 5.0 | 41.0 | 101.0 | 191.0 | 294.0 |
| | 2.0 | 234550.0 | 96.857510 | 70.261685 | 5.0 | 31.0 | 88.0 | 164.0 | 277.0 |
| | 3.0 | 215878.0 | 110.374698 | 77.998615 | 5.0 | 50.0 | 73.0 | 160.0 | 294.0 |
| | 4.0 | 386375.0 | 61.472551 | 53.699586 | 5.0 | 31.0 | 34.0 | 77.0 | 294.0 |
| International e-commerce | 1.0 | 535433.0 | 112.804614 | 73.024885 | 5.0 | 56.0 | 100.0 | 164.0 | 300.0 |
| | 2.0 | 746934.0 | 78.712368 | 69.273796 | 5.0 | 27.0 | 55.0 | 118.0 | 300.0 |
| | 3.0 | 989450.0 | 56.765560 | 31.572350 | 5.0 | 30.0 | 50.0 | 70.0 | 282.0 |
| Milk powder | 1.0 | 230146.0 | 92.658226 | 77.302621 | 5.0 | 30.0 | 63.0 | 139.0 | 294.0 |
| | 2.0 | 230986.0 | 86.187946 | 68.891675 | 5.0 | 25.0 | 65.0 | 136.0 | 277.0 |
| | 3.0 | 202042.0 | 103.748008 | 75.470011 | 5.0 | 50.0 | 70.0 | 160.0 | 294.0 |
| Mobile e-commerce app install | 0.0 | 249768.0 | 62.984858 | 60.692729 | 5.0 | 18.0 | 41.0 | 88.0 | 277.0 |
| Oil | 1.0 | 552161.0 | 104.898805 | 68.215862 | 5.0 | 54.0 | 89.0 | 144.0 | 267.0 |
| | 2.0 | 719135.0 | 78.075060 | 64.659368 | 5.0 | 28.0 | 57.0 | 119.0 | 267.0 |
| | 3.0 | 793670.0 | 68.295494 | 31.518908 | 11.0 | 55.0 | 77.0 | 77.0 | 265.0 |
| Software | 1.0 | 725848.0 | 108.542180 | 68.110905 | 5.0 | 60.0 | 95.0 | 149.0 | 267.0 |
| | 2.0 | 422152.0 | 78.643958 | 64.401920 | 5.0 | 29.0 | 58.0 | 118.0 | 267.0 |
| | 3.0 | 236499.0 | 70.135882 | 22.241696 | 11.0 | 55.0 | 77.0 | 77.0 | 265.0 |
| Telecom | 1.0 | 170527.0 | 97.856199 | 76.078652 | 5.0 | 41.0 | 71.0 | 143.0 | 294.0 |
| | 2.0 | 195532.0 | 81.011456 | 69.580404 | 5.0 | 16.0 | 62.0 | 142.0 | 277.0 |
| | 3.0 | 171075.0 | 97.894175 | 83.078851 | 5.0 | 20.0 | 70.0 | 160.0 | 294.0 |
| Tire | 1.0 | 444487.0 | 97.604591 | 65.354584 | 5.0 | 48.0 | 85.0 | 133.0 | 267.0 |
| | 2.0 | 528118.0 | 72.866954 | 62.683423 | 5.0 | 25.0 | 49.0 | 107.0 | 266.0 |
| | 3.0 | 594824.0 | 72.237033 | 42.437838 | 5.0 | 52.0 | 73.0 | 84.0 | 265.0 |

Table 4.1: Distribution of strategy scores in the iPinYou dataset grouped by our chosen context features: "advertiser category" and "ad exchange".

| advertiser_category_context | adexchange_context | region | os | browser | gender | slotvisibility | slotformat | slotwidth | slotheight | score |
|---|---|---|---|---|---|---|---|---|---|---|
| Footwear | 4.0 | 374 | windows | firefox | Female | FirstView | Na | 300 | 250 | 120.000000 |
| Software | 2.0 | 146 | windows | theworld | Male | 1 | 0 | 728 | 90 | 81.625000 |
| Telecom | 1.0 | 65 | mac | other | Unknown | FirstView | Na | 300 | 250 | 80.000000 |
| Footwear | 3.0 | 0 | windows | chrome | Unknown | Na | Na | 300 | 250 | 127.613537 |
| Footwear | 4.0 | 216 | windows | safari | Female | SecondView | Na | 300 | 250 | 36.000000 |

Context $c_i$ Strategy $s_i$ utility score $\mu_i$

Figure 4.4: Context features, strategy features, and utility score on a sample from the iPinYou Dataset.

two context features, and "Cost per Click" as the utility score. Resulting in only 785K possible combinations, or which only 8K are present in the training dataset due to the relatively low diversity of our advertisers' profiles.

A sample of the iPinYou dataset is shown in Figure 4.4 and shows the different types of features.

#### 4.4.1.2 Evaluation Protocol

Assessing our approach to contextual advertising strategy generation is challenging, similar to other generative tasks, because we cannot simply rely on direct comparisons with a test dataset. Our model receives only context features as input and is trained to either replicate successful strategies from the training dataset or to generate new successful ones. Consequently, during testing, we need to evaluate each generated strategy against the entire test set rather than merely comparing it to a corresponding row in the dataset. To determine the quality and trustworthiness of the strategies we generate, we assess their effectiveness (positiveness) and their similarity to strategies within the test dataset.

The dataset is split into training, validation, and test sets. But the evaluation protocol on the test set is different than usual. For each unique context in the test set, we generate up to $k = 300$ unique strategies to simulate the structure of a complete advertising campaign with multiple strategies.

We then compare the set of generated strategies to the test dataset and to other sets generated by the other methods using the evaluation metrics.

The goal is to evaluate the capacity of each model to generate high-performing strategies and its ability to generate strategies that are closer to ground truth data while also being diverse.

#### 4.4.1.3 Evaluation metrics

To quantify the quality of a set of generated strategies for a given context, we measure four different metrics:

- **Mean utility score** accross the generated strategies set with:

$$\overline{u_c^G} = \frac{1}{\left|\widehat{S_c}\right|} \sum_{s_c^G \in \widehat{S_c}} E\left(c, s_c^G\right) \tag{4.14}$$

- **Mean hamming distance** between the generated strategies and the best strategies in the training set. To measure the ability of the model to replicate the best strategies from the dataset. Defined as follows:

$$h\left(s^G, s\right) = \frac{1}{n_S} \sum_{i=1}^{n_S} 1_{s_i^G = s_i} \tag{4.15}$$

$$\overline{h_c^G} = \sum_{s_c^G \in \widehat{S_c}} \max_{s_c \in S_c^*} h\left(s_c^G, s_c\right) \tag{4.16}$$

- **Mean cosine distance** between the generated strategies and the best strategies in the training set. Calculated feature wise on the embedding vectors from the Estimator. To measure if the generated strategies are close to the best ones in the training dataset even if they differ (e.g. if the country is different but has no real impact on the performance)

$$sim\left(s^G, s\right) = \frac{1}{n_S} \sum_{i=1}^{n_S} \frac{emb_E\left(s_i^G\right) . emb_E\left(s_i\right)}{\left\|emb_E\left(s_i^G\right)\right\|_2 . \left\|emb_E\left(s_i\right)\right\|_2} \tag{4.17}$$

$$\overline{sim_c^G} = \sum_{s_c^G \in \widehat{S_c}} \max_{s_c \in S_c^*} sim\left(s_c^G, s_c\right) \tag{4.18}$$

- **Overall diversity** being the number of generated strategies by context divided by $k$

$$diversity\left(\widehat{S_c}\right) = \frac{\left|\widehat{S_c}\right|}{k} \tag{4.19}$$

### 4.4.2 Comparison Models: parameters and info

To the best of our knowledge, our method is the first method to truly satisfy all the imposed constraints of the contextual advertising strategy problem. So, in the absence of directly comparable methods, we evaluate the performance of our proposed ASGAR framework to three classes of baseline methods (see Section 2.2.3):

- **Contextual Multi-Armed Multi-Agent Bandits:** as described in [Ban et al., 2021], with centralized and decentralized approaches.
- **Genetic Evolutionary Model:** on various population sizes with the fitness function being the Estimator.
- **Auto-Regressive Contextual Conditional Generative Model:** as described in [Wei et al., 2022b], where strategy features are converted to sequences and the model generates the next token based on a condition that dictates which feature to generate.

We also compare two versions of our model: a **high fidelity** (low exploration) version in which we trained our model to generate strategies similar to the training data, and a **low fidelity** (high exploration) version in which we trained it to explore more and generate unseen strategies.

We show in Table 4.2 the constraints that are satisfied by each model.

Table 4.2: Constraints satisfied by the compared models.

| | Genetic Model | Multi-Agent Multi-Armed Bandits | Auto-Regressive Generative Model | ASGAR (Ours) |
|---|---|---|---|---|
| **Non Combinatorial** | Exploratory | Yes | Yes | Yes |
| **Order Agnostic** | Yes | Yes | No | Yes |
| **Parametrizable Data Fidelity** | No | Some | Some | Yes |
| **Feature Interaction Aware** | Yes | No | Yes | Yes |
| **Utility Aware** | Yes | Yes | No | Yes |

To generate multiple strategies for the same context, we either keep the dropout modules active during the inference of the compared models, use the variational part of the model, or directly sample from the forecasted feature value distributions. Keeping the dropout active simulates the training of multiple models where each one converges to a different plausible solution.

### 4.4.3 Implementation Details

For fair comparisons, we implement all the models with PyTorch[1] and optimize all the models with Adam [Kingma and Ba, 2015]. The embedding size for the Estimator is set to 16, in our study we used DCNv2 [Wang et al., 2021a] as our Performance Estimator because it is currently one of the best models for modeling feature interactions and because it is easier to train than newer transformer-based models as in [Dilbaz and Saribas, 2023, Song et al., 2019]. We adapted the output of the model to make it a regression method rather than a click classification model.

The batch size is fixed to 16. The learning rate is scheduled with a Cyclic Scheduler with "triangular2" mode between 0.0001 and 0.01. For ASGAR, we use the structure (embedding size=32, transformer heads=16, transformer layers=32, dropout=0.5). For the genetic method, we use (population=1000, iterations=1000, elite ratio=0.1, mutation probability=0.5, parent crossing=0.3). For the Multi-Armed Bandits we use (layers=3, hidden size=64, $\delta = 0.1$,$\lambda = 1$). For the autoregressive model, we use (layers=4, hidden

[1] Code available at `https://github.com/IssamBenamara/ASGAR`

Table 4.3: Comparison of the evaluation metrics over the results of each model

| | Mean Score ↑ | Mean Hamming | Mean Cosine | Mean Diversity ↑ |
|---|---|---|---|---|
| **Genetic Model** | 155.73 | 0.652 | 0.520 | **1.00** |
| **Decentralized Multi-Agent Multi-Armed Bandits** | 127.71 | 0.530 | 0.584 | 0.513 |
| **Centralized Multi-Agent Multi-Armed Bandits** | 112.81 | 0.564 | 0.652 | 0.492 |
| **Auto-Regressive Generative Model** | 95.19 | 0.143 | 0.925 | 0.163 |
| **ASGAR (Ours) High Fidelity** | 133.51 | 0.235 | 0.797 | 0.816 |
| **ASGAR (Ours) Low Fidelity** | **170.41** | 0.545 | 0.593 | **1.000** |

size=128, $\tau = 1$).

All activation functions are ReLU and the dropout rate is 0.5. We perform early stopping according to the utility score estimation of the validation set. Each experiment is repeated 10 times, and the average results are reported. For the autoregressive model, we test different orderings of the features and keep the best results.

In ASGAR, we set the hyperparameters to: $ne = 1, nf = 2, pdf = 0.1, pei = 1, aei = 0.5, aa = 1$.

### 4.4.4 Results

In this section, we compare the performance of ASGAR with the baseline methods. Table 4.3 shows the experimental results of all compared models over the iPinYou dataset.

The Estimator DCNv2 achieves great results in the forecasting task with a mean absolute percentage error equal to 15% on the iPinYou dataset. The *score* and *diversity* metrics are the only monotonic metrics during evaluation and can be interpreted as good or bad performance. The other metrics, *Cosine* and *Hamming*, assess the similarity to historical data (data fidelity) to evaluate how closely the output aligns with established historical patterns. For example, a high *Cosine* value or a low *Hamming* value suggests that the generated strategies closely resemble the successful strategies in the dataset. Data fidelity is a manageable aspect, and its interpretation—whether the values are good or bad—depends entirely on the model's intended behavior, whether focusing on exploration or exploitation.

The autoregressive method is observed to perform poorly in terms of utility score relative to other models, yet it exhibits the highest data fidelity with minimal diversity. This combination of high fidelity and low diversity suggests that while the autoregressive method excels at generating plausible strategies, its inherent structure may limit its ability to explore a broader range of potentially successful strategies. This limitation can be somewhat mitigated by introducing sampling to enhance diversity, though this may compromise data fidelity. The performance characteristics of the autoregressive method underscore the advantage of adopting non-autoregressive models for better overall effectiveness. Multi-Armed Bandits, while achieving relatively good scores, face challenges with data fidelity. This is due to the independent sampling of each feature arm during generation, which fails to consider interactions between features. The Genetic Model being a combinatorial exploration method is able to achieve great scores but has the worst data fidelity compared to the other methods because of its noncontrollable exploration/exploitation rate. Figure 4.5 compares the utility score distribution of the strategies generated by ASGAR and the genetic model, showing that while the genetic model is able to reach higher scoring strategies, it also generates many low performers. Whereas, ASGAR's generated strategies are more centered around a high score.

ASGAR outperforms all baselines in its low-fidelity version in terms of utility score. While its high-fidelity version is surpassed by the Genetic Method, it manages to have higher data-fidelity. This makes it more trustable than the fully exploratory Genetic

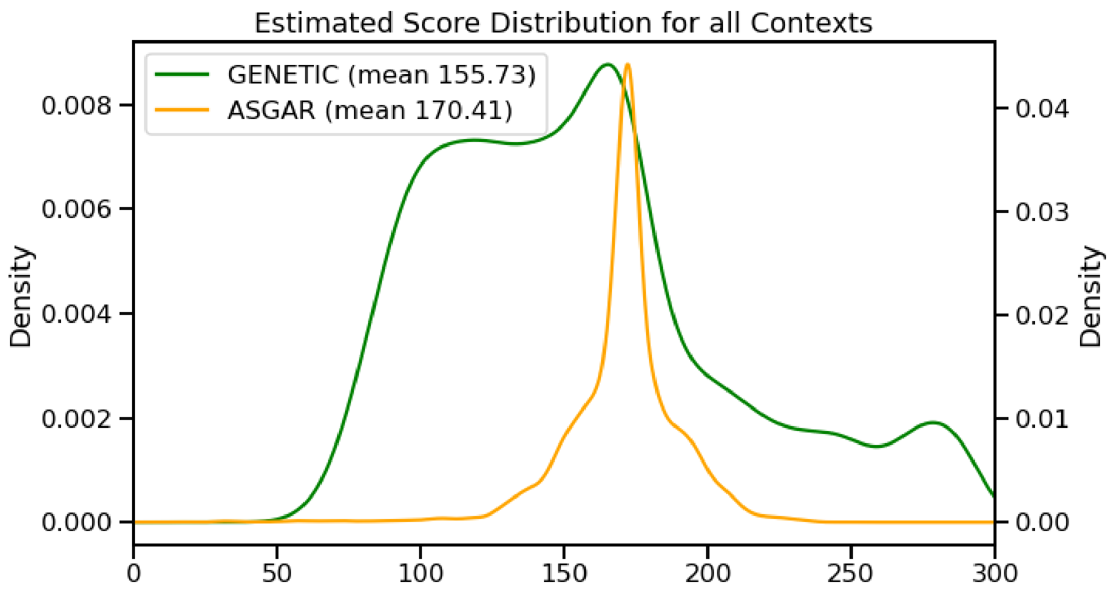



Figure 4.5: Comparison of two distributions of the generated utility scores between the Genetic model and ASGAR.

Method.

The Auto-Regressive Method achieves the best data fidelity metric in our experiments, but this behavior strongly decreased its generative diversity with only 0.163 diversity. Meaning that on average, for each context it generates around 49 strategies instead of the $k = 300$ required strategies.

We experimented with various $k$ values to simulate different sized advertising campaigns. As shown in Figure 4.6, our model outperforms the genetic model in high values of $k$. However, even if for small $k$ values the genetic model is the best, ASGAR outperforms it in data-fidelity. Therefore, from a business standpoint, it is more reliable to use ASGAR as it provides fewer risks with relatively minimal loss in score.

Figure 4.7 shows that ASGAR learns to reconstruct positive strategies better than negative strategies. This proves that the interpolation in the loss function is effective as a

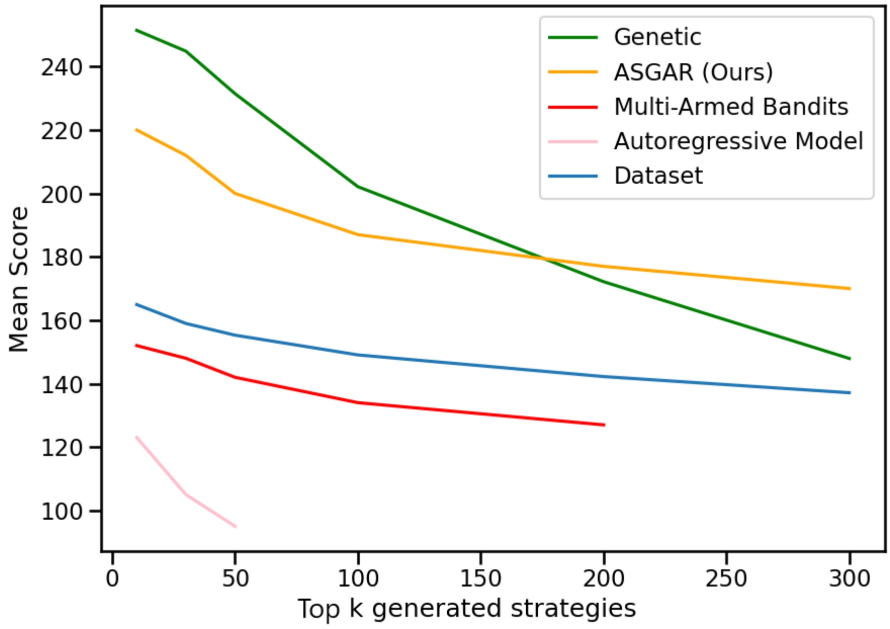



Figure 4.6: Generated scores for different top $k$ values, showing robustness and diversity of each model.

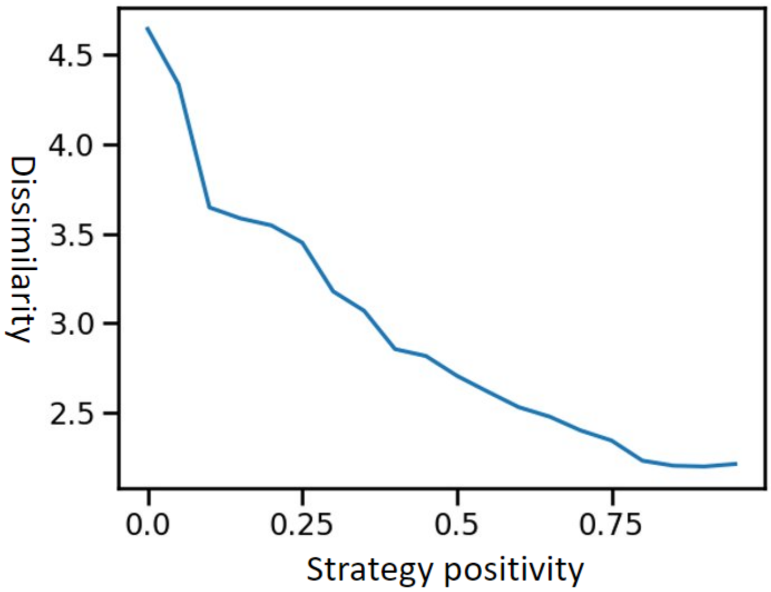



Figure 4.7: Generated strategy similarity to dataset strategies relative to their positiveness.

smooth contrastive learning approach.

### 4.4.5 Deployement

ASGAR is currently in production for The Programmatic Company. It has shown very promising results even while trained on only 8K unique strategies of the private dataset (compared to 155K for iPinYou). The advertising strategies generated by ASGAR for The Programmatic Company have demonstrated impressively low Cost-Per-Click rates, highlighting its effectiveness and robustness even when limited to smaller datasets.

A new version of ASGAR is under development, leveraging a more extensive dataset that comprises over 200K unique strategies, enriched with 12 strategy features and 2 context features. This expansion is aimed at enhancing the model's complexity and predictive power. Upcoming studies will focus on evaluating this enhanced model's performance in comparison to that of human experts, aiming to quantify improvements and identify areas for further optimization.

### 4.4.6 Complexity and Ablation Analysis

We performed complexity and ablation studies on ASGAR to test the efficiency of its architecture. Ablation is a systematic method of assessing the contribution of individual components of a model by selectively removing or modifying them and observing the effect on the model's performance. As shown in Table 4.4, our model is robust and is able to learn on the iPinYou dataset on the different architectures we tested without losing meaningful performance. The same can be observed when varying the number of strategy features. However, we also note that its number of parameters grows rapidly and requires a lot of GPU resources to run.

We performed two ablation analyses: no attention mechanism and no Estimator guidance. Figure 4.8 shows that removing the attention mechanism by replacing it with

Table 4.4: Various architecture parameters and resulting scores

| | | | | | | |
|---|---|---|---|---|---|---|
| **Embedding Size** | 4 | 8 | 32 | 32 | 128 | 128 |
| **Transformer Heads** | 4 | 8 | 8 | 16 | 8 | 32 |
| **Transformer Layers** | 3 | 4 | 64 | 32 | 4 | 128 |
| **Score** | 116.97 | 138.25 | 163.06 | 170.41 | 162.42 | 165.28 |

a multi-layer perceptron makes ASGAR fail to learn at all by not being even able to converge even when trying multiple values of learning rate. Metrics are irrelevant in this context as the model has not exhibited convergence.

When removing the Estimator's guidance the model struggles to converge, thus showing the same behavior as in Figure 4.8. However, it manages to learn a few good strategies, as it generates better scores than when removing the attention mechanism. This is because it is focused only on exploitation with no exploration. The model tries to strictly learn all the positive strategies and avoid negative strategies, but because of the nature of the dataset, some positive strategies are very similar to negative ones (sometimes a single feature change can lead to very different performance), which makes the learning process unstable. Whereas if the Estimator is enabled, the model is guided in its avoidance of negative strategies and thus proposes a novel promising strategies via exploration.

### 4.4.7 Discussion

Although ASGAR outperforms the other baseline methods while also satisfying all the imposed constraints, it has the following notable limitations:

- The rapid growth in number of parameters could make it costly to train.
- Relying on the quality of the Estimator for exploration and evaluation means that


Figure 4.8: Train and validation losses unable to decrease with attention ablation despite the cyclic learning rate scheduler.

the Estimator needs to have high accuracy to avoid misleading the Generator.

- The strategy dissimilarity function we used is not well adapted to the problem. Because a single feature change could lead to a big utility change while the dissimilarity score remains approximately the same. The desired behavior would be that a change in an important feature would lead to a significant change in the dissimilarity value.
- Risks of mode collapse. Since we use a variational part along with keeping the dropout active, we were able to mitigate the risks of mode collapse. However, the risk is still present.

Table 4.5 presents an instance of severe mode collapse we occasionally encountered, where our model outputs the same high scoring strategy for all the contexts. This issue is observed with the use of smaller embedding dimensions or, more broadly, when diminishing the model's capacity by decreasing its parameter count. This challenge, combined with the well-documented difficulties associated with variational methods and their propen-

| | advertiser_category_context | adexchange_context | region | os | browser | gender | slotvisibility | slotformat | slotwidth | slotheight | generated_score | real_mean_score |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Chinese vertical e-commerce | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 174.569534 | 68.504595 |
| **1** | Chinese vertical e-commerce | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 144.019531 | 79.143634 |
| **2** | Chinese vertical e-commerce | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 151.652618 | 62.169752 |
| **3** | Footwear | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 169.081909 | 118.654785 |
| **4** | Footwear | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 148.299820 | 96.857510 |
| **5** | Footwear | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 163.155609 | 110.374698 |
| **6** | Footwear | 4.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 145.341019 | 61.472551 |
| **7** | International e-commerce | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 180.010086 | 112.804614 |
| **8** | International e-commerce | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 148.775879 | 78.712368 |
| **9** | International e-commerce | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 156.986557 | 56.765560 |
| **10** | Milk powder | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 167.448547 | 92.658226 |
| **11** | Milk powder | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 144.110489 | 86.187946 |
| **12** | Oil | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 167.926117 | 104.898805 |
| **13** | Oil | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 146.054123 | 78.075060 |
| **14** | Oil | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 148.415695 | 68.295494 |
| **15** | Software | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 176.825775 | 108.542180 |
| **16** | Software | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 148.351486 | 78.643958 |
| **17** | Software | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 145.190353 | 70.135882 |
| **18** | Telecom | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 180.883865 | 97.856199 |
| **19** | Telecom | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 154.234650 | 81.011456 |
| **20** | Telecom | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 155.129440 | 97.894175 |
| **21** | Tire | 1.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 176.444473 | 97.604591 |
| **22** | Tire | 2.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 154.675720 | 72.866954 |
| **23** | Tire | 3.0 | 0 | 0 | 0 | 1 | 3 | 3 | 1 | 1 | 148.495804 | 72.237033 |

Table 4.5: Extreme case of mode collapse. ASGAR outputs the same high scoring strategy for all contexts.

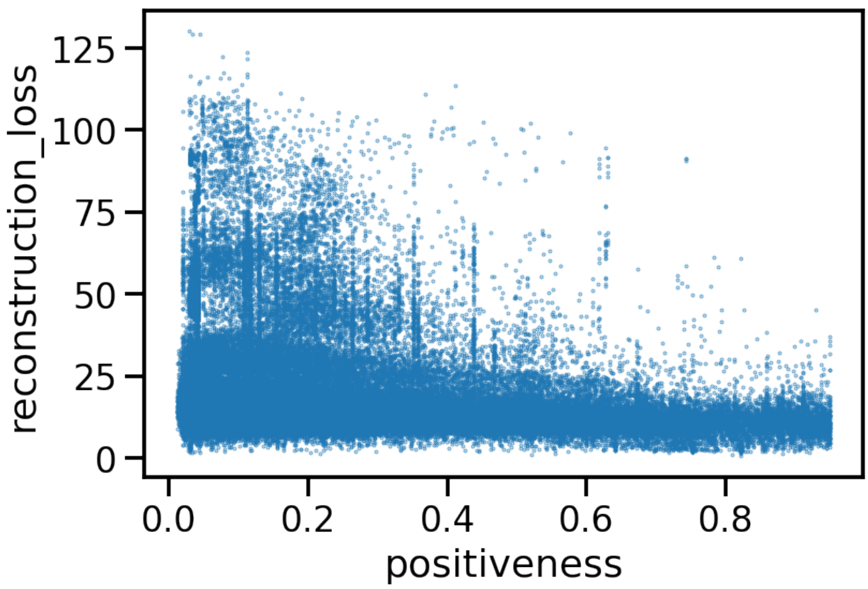



Figure 4.9: Instance level dissimilarity between generated strategies and the original strategies.

sity for mode collapse [Takida et al., 2022, Liu et al., 2023a], rendered the training of our model a delicate task. Furthermore, this situation underscores the limitation of relying solely on the Estimator for exploratory purposes. Without any mechanism to enforce diversity, the current loss function could misleadingly suggest that continuously generating the same high-scoring strategies is a viable approach, although it clearly does not suit all situations. Specifically, in this instance, the variational component of our model tends to converge to parameters of a normal distribution with zero mean and zero variance, effectively nullifying the impact of the variational part and leading it to output the same high-scoring strategies consistently.

The average similarity of generated strategies illustrated in Figure 4.7 demonstrates a noticeable trend towards better reconstruction of positive strategies. However, an examination at the individual instance level, as depicted in Figure 4.9, reveals that numerous negative strategies are also being reconstructed. This issue stems from the presence in the dataset of many strategies that vary by only one or two features yet yield vastly different outcomes. This is exacerbated by our current strategy dissimilarity function, which is not well adapted for changes in single features.

## 4.5 Conclusion

In this chapter, we detailed our approach to tackling the intricate challenges inherent in digital advertising strategy design, specifically addressing four of the five critical constraints we previously outlined as essential for success in this domain.

We proposed a novel framework for contextual advertising strategy generation named ASGAR along with a novel loss function for guided feature space exploration. ASGAR leverages the attention mechanism and smooth contrastive learning to generate advertising strategies in a non-combinatorial and non-autoregressive fashion while offering parameterizable generative fidelity to the training dataset. To the best of our knowledge, this is the first method to satisfy the main imposed constraints of the advertising strategy generation problem which lets it outperform the compared methods. We discuss its advantages and limitations and present extensive experimental results on a public dataset to demonstrate its efficacy. Our results demonstrated superior results, outperforming other approaches while adhering to the task's constraints and providing adjustable exploration/exploitation options essential for meeting diverse client needs.

Our contributions within this chapter serve as the building blocks for the next chapters where we explore and address several future directions initially identified in the published paper. In Chapter 5, we address the limitations related to mode collapse, inadequate dissimilarity function and add another exploration mechanism. In Chapter 6, we aim to satisfy the fifth constraint of controllable generation.

# Chapter 5

# Generation Diversity

## Leveraging Quantization for Controllable Diversity and Exploration in Advertising Strategy Generation

**Contents**

This chapter is a follow up to Chapter 4. We address some of the previously identified limitations to improve performance and diversity, and propose an enhanced evaluation protocol. We introduce a learned metric network addressing a limitation in *ASGAR*'s dissimilarity function to ensure sensitivity to minor feature changes, thereby reducing confusing signals during training, and boost overall performance. We propose a quantization process, which efficiently mitigates the risks of mode collapse (see Section 4.4.7), improves diversity, and introduces an inference-time exploratory mode. We evaluate the results of our proposals and measure the uplift in performance compared to our previous methods.

The rest of this chapter is organized as follows: We start with an introduction in Section 5.1. In Section 5.2 we discuss related work. In Section 5.3.1 we detail the *Aligner*, a learned metric network that we propose to disentangle strategy representations. In Section 5.3.2 we detail the quantization process. In Section 5.4, we evaluate our approaches, measure their impact on *ASGAR*, and discuss their limitations. Finally we provide a conclusion in Section 5.5.

## 5.1 Introduction

It is essential for marketers to have control over the level of exploration in advertising strategy generation to manage investment risks effectively while uncovering new and effective advertising strategies. This necessity ranges from desiring the generation of outputs that are reliable and closely mirror the ground truth data, to seeking outputs that are more exploratory or creative, thus navigating the delicate balance between exploration and exploitation.

Traditionally, in the realm of generative and recommendation tasks, initial guidance is provided by conditioning the model on contextual inputs [Mirza and Osindero, 2014, Zhang et al., 2021a, Ilias and Askounis, 2023, Nichol et al., 2022, Patashnik et al., 2021]. Subsequently, various techniques are employed to diversify the outputs, such as sampling methods [Fan et al., 2018, Holtzman et al., 2020, Keskar et al., 2019]. For instance, in text generation, the auto-regressive approach might involve selecting a less probable next word, which, while less likely, can still produce a high-quality text overall.

However, in the specialized case of digital advertising strategy generation, and as shown in Chapter 4, we can't use sampling methods for optimal exploration as it immediatly breaks the complex features interactions and the cohesiveness of the generated strategy. In this chapter, we build upon our previous work *ASGAR* [Benamara and Viennet, 2023] for contextual advertising strategy generation, enhancing its performance and introducing a mechanism for exploratory generation at inference time. Our principal contributions include:

- We enable an inference-time exploratory mode, facilitated by the incorporation of vector quantization as a variational component. This addition not only helps in stabilizing the model against mode collapse (see Section 4.4.7), but also importantly introduces a mechanism for exploration during inference.
- We mitigate a significant limitation previously identified, through the develop-

ment of a learned metric neural network that we call the *Aligner*. This network, integrated with *ASGAR*'s loss function, refines the model's ability to discern and differentiate between strategies during the reconstruction process, which reduces confusing signals and enhances strategy uniqueness.

- We propose an improved evaluation protocol, complete with novel metrics designed to assess the enhancements made to the model more accurately during and after training.

We evaluate our approach using the same iPinYou dataset [Liao et al., 2014], enabling a direct comparison with our prior work and quantifying the improvements achieved. This chapter thus outlines our advancements in offering a more reliable diversification and flexibility in the generation of digital advertising strategies.

## 5.2 Related Work

### 5.2.1 Metric Learning

Learned metrics, often derived from trained neural network models, have become a cornerstone in advancing the field of machine learning, particularly in tasks requiring nuanced understanding of data similarity and dissimilarity. These metrics are designed to transcend traditional distance measures by encapsulating complex patterns and relationships within data, tailored for specific tasks.

The objective of Metric Learning is to devise a representation function that positions objects within an embedding space, where the spatial relationships preserve the similarities between objects — ensuring that similar items are positioned closely while dissimilar ones are placed further apart. A variety of loss functions have been introduced to achieve this objective. For instance, contrastive loss [Chopra et al., 2005] encourages the model to map objects of the same category to identical points and objects from distinct categories to points that are separated by a margin greater than a specified threshold. Another widely used approach is the triplet loss [Schroff et al., 2015a], which aims to ensure that the distance between an anchor sample and a positive sample (similar) is less than the distance between the anchor sample and a negative sample (dissimilar). Other works extended to the case with one positive example and multiple negative examples [Sohn, 2016, Radford et al., 2021].

Learned metrics networks can be used in two ways, depending on the architecture and the specific goals of the training process:

- **Gradients are propagated:** In some cases, learned metrics are an integral part of the model's training process and the gradients from the learned metric loss are propagated back through the network, allowing the model to adjust its parameters to improve the metric.
- **Gradient are NOT propagated:** In other scenarios, learned metrics are used as a separate component to evaluate or guide the training process without directly influencing the training of the main model (e.g. reinforcement learning).

In many studies [Kaya and Bilge, 2019, Sung et al., 2018, Schroff et al., 2015b], learned metrics are applied to image data, facilitating seamless gradient propagation since the output of the training model can be directly fed into the learned metric network without interrupting the flow of gradients. However, in our situation, our model produces soft-maxed logits corresponding to multiple features. This effectively results in the selection of indices through an argmax operation, a non-differentiable process that interrupts the gradient flow.

To address the issue of disrupted gradient flow caused by non-differentiable operations like argmax selection, several strategies have been developed. One popular approach is the use of the Gumbel-Softmax trick [Jang et al., 2017], a differentiable approximation to sampling discrete variables that allows for the backpropagation of gradients through discrete choices. Another approach is the Straight-Through (ST) estimator [Bengio et al., 2013, Cheng et al., 2019], which allows gradients to pass through the discrete variable unchanged during the backward pass. Both of these methods offer practical solutions for maintaining gradient flow in models that involve discrete outputs, thereby enabling the integration of learned metrics without compromising the model's ability to learn through backpropagation. These techniques have been pivotal in extending the application of learned metrics beyond continuous data domains, facilitating their use in more complex models that output discrete variables.

### 5.2.2 Vector Quantization Methods

Vector quantization has emerged as a pivotal technique in both data compression and representation learning, garnering particular attention in models such as VQ-VAE [van den Oord et al., 2017]. This method involves mapping input data onto a limited set of vectors, effectively quantizing the input space. VQ-VAE [van den Oord et al., 2017, Razavi et al., 2019] introduced a commitment loss and utilized Exponential Moving Average (EMA) to enhance codebook learning. Building upon this foundation, VQ-GAN [Esser et al., 2021] extended the application of VQ-VAE within its autoencoder framework. It decoded from a quantized representation and subsequently trained a transformer-style model to generate novel quantized representations, which were then decoded into new instances of data. This approach has seen adaptation and expansion in various models, including those employing auto-regressive and masked generation techniques [Chang et al., 2022, Kolesnikov et al., 2022], leading to improved efficiency and adaptability in deep learning architectures.

Further developments in vector quantization aimed to either boost performance or mitigate limitations inherent in the codebook method. Finite Scalar Quantization (FSQ) [Mentzer et al., 2023] adopts bounded scalar quantization to compress codes into scalar values, achieving quantization by rounding to the nearest scalar. RQ-VAE and RQ-

Transformer [Lee et al., 2022] utilize residual quantization, enhancing the precision of quantized codes by encoding and storing quantized residuals. The authors in [El-Nouby et al., 2023] proposed the concept of product quantization, which divides the codebook into a series of smaller codebooks, allowing for more complex and nuanced data representation. These innovations underscore the evolving landscape of vector quantization, highlighting its crucial role in the advancement of generative modeling and data compression techniques.

## 5.3 Methodology

### 5.3.1 Improved Reconstruction via a Learned Metric Backpropagation

In our prior work, we employed a strategy dissimilarity function that was equal to the sum of cross-entropy losses across each strategy feature. While this function proved to be satisfactory to train *ASGAR* [Benamara and Viennet, 2023], it was not ideal because it inherently lacks a mechanism to weigh the importance of different features. Therefore, when two strategies differ in a single feature, but that single change resulted in a significant difference in performance, they still received nearly identical dissimilarity scores which is not the desired behavior.

Ideally, the goal is to have a dissimilarity function $R$ that scores based on the significance of the differing features. The dissimilarity score should closely reflect the difference in features as well as the difference on performance. Given three strategies $s_1^+$, $s_2^+$ and $s_3^-$ such that the three differ in only a single feature from each other, but $s_1^+$ and $s_2^+$ yield approximately the same high performance score and $s_3^-$ yields a low performance score. The dissimilarity function $R$ should then follow this inequality:

$$R\left(s_1^+, s_2^+\right) \ll R\left(s_1^+, s_3^-\right) \approx R\left(s_2^+, s_3^-\right) \tag{5.1}$$

Rather than conducting a feature importance analysis and individually adjusting the weight of each term in the summation of cross-entropy losses to enhance this aspect, we observed that the *Estimator* network is sensitive to feature swaps depending on their impact on the performance of a strategy. This sensitivity arises from the *Estimator* being a state-of-the-art model capable of capturing complex interactions between features. This means it can accurately assess the impact of altering a single feature on overall performance. Therefore, we propose to leverage the *Estimator*'s latent projection (the weights from the final layer before regression) as a richer representation of a strategy to refine the dissimilarity function.

Specifically, as shown in Figure 5.1 we start by constructing a dataset derived from the original dataset. For each strategy, we identify all strategies that differ from it by $k_{diff}$ features or fewer. Among these, strategies that exhibit a performance difference greater than $u^-$ percent are categorized as negative examples, while those with a performance difference of less than $u^+$ percent are classified as positive examples. Each example of the dataset is a triplet defined by:

$$\left(s, S_s^+ = \left\{s' \in S, s' \approx s \wedge |u_{s'} - u_s| < u^+\right\}, S_s^- = \left\{s' \in S, s' \approx s \wedge |u_{s'} - u_s| > u^-\right\}\right) \tag{5.2}$$

Then, we design a neural network model that we name the *Aligner* (as it aligns strategy representations) of which the base layers are frozen layers taken from the *Estimator* and subsequent layers are trainable. The subsequent layers are projection heads, followed by a regression layer that outputs a similarity score. This network will serve as a learned metric model. Training is conducted similarly to the CLIP approach [Radford et al., 2021], where we aim to bring the strategy and its positive examples closer together by maximizing their cosine similarity. Conversely, we aim to distance the negative samples by minimizing their cosine similarity. The cosine similarity is measured on the projected representations learned by the *Aligner*.

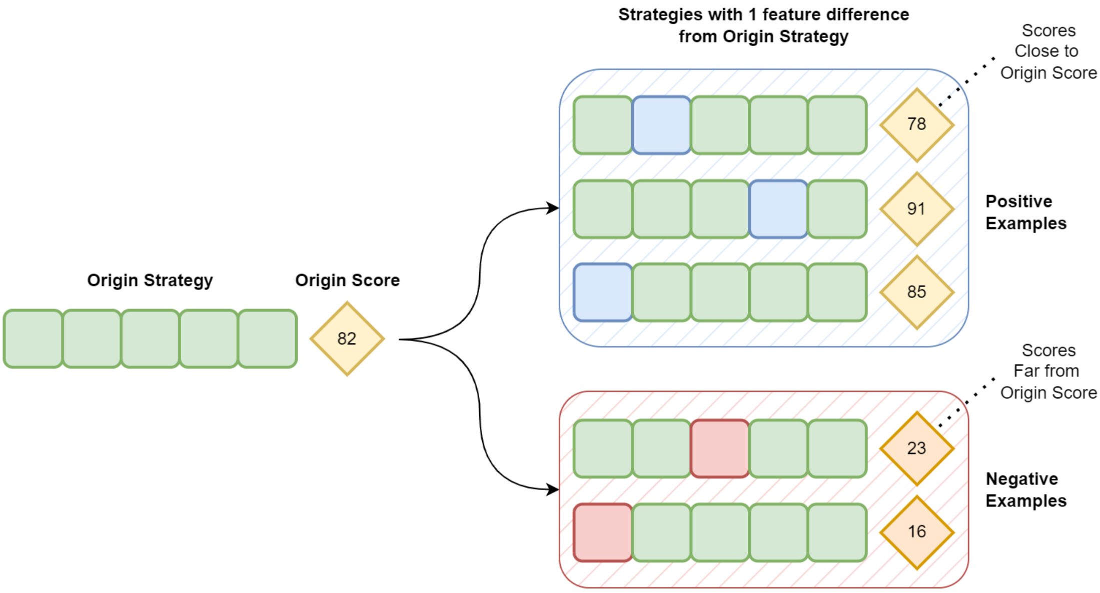



Figure 5.1: Dataset preparation for metric learning. For each strategy, similar strategies are split into positive and negative examples based on their performance Score.

Figure 5.2 shows a simplified view of the architecture of the *Aligner* and how it is trained using a single positive example at a time for each strategy and considering the rest of the batch strategies as negative examples. During inference, the *Aligner* is given a pair of strategies, it projects them into its own latent space through which we can directly measure their cosine similarity.

Once the *Aligner* model is trained, it can either augment or entirely replace the existing dissimilarity function during *ASGAR*'s training phase. As will be shown in Section 5.4 on experimental results, a gradual shift from the straightforward Cross-Entropy sum approach to the intricate learned metric network proves to be more effective. This phased integration aids *ASGAR*'s learning process by initially steering it towards a general direction, then by progressively incorporating the *Aligner*—either as an additive component in the loss equation, as a weighting to Cross-Entropy sum, or by fully replacing it—the system fine-tunes *ASGAR*'s representations to capture more subtle differences. Equation 5.3 shows a type of possible scheduling where $\tau$ is a temperature scaling parameter to amplify the effect of the *Aligner*'s cosine similarity score, $t$ is the timestep

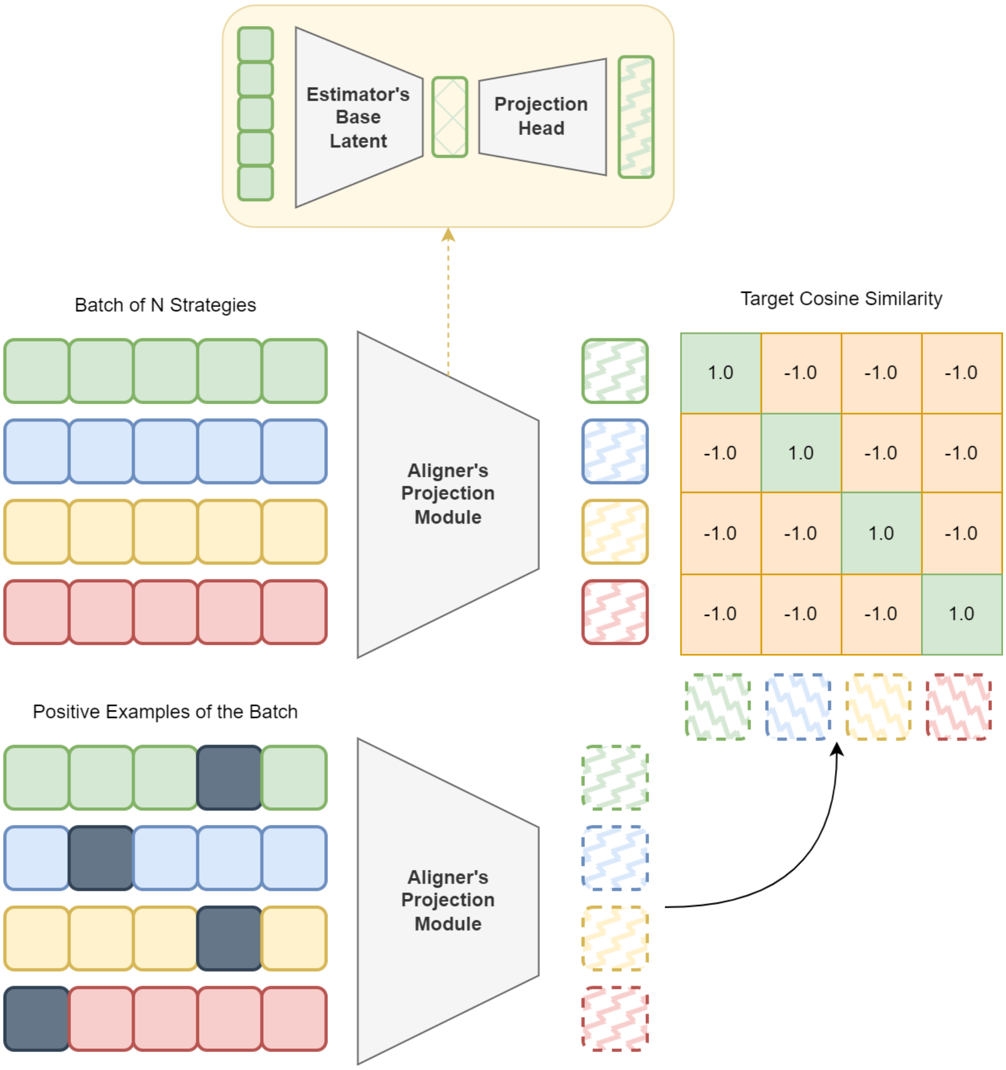



Figure 5.2: *Aligner* CLIP-like [Radford et al., 2021] contrastive training.

or epoch during training, $T_1$ and $T_2$ are predefined timesteps to transition from one dissimilarity function to the other. The term $((Aligner_{cosine-similarity}(s, \hat{s}) - 1)^{2.\tau} + 1)$ maps the values such that the value 1 means perfect similarity and higher values means more dissimilarity.

$$R(s, \hat{s}, t) = \begin{cases} CE(s, \hat{s}) & t < T_1 \\ ((Aligner_{cosine-similarity}(s, \hat{s}) - 1)^{2.\tau} + 1) .CE(s, \hat{s}) & T_1 \leqslant t < T_2 \\ (Aligner_{cosine-similarity}(s, \hat{s}) - 1)^{2.\tau} & T_2 \leqslant t \end{cases} \tag{5.3}$$

A significant issue emerges when integrating the *Aligner* with *ASGAR*: the *Aligner*'s initial layer is an embedding layer, whereas *ASGAR* produces logits that are transformed into indices via an argmax operation. This nondifferentiable operation interrupts the flow of gradients from the *Aligner* back to *ASGAR*, diminishing or even stopping the *Aligner*'s effectiveness in steering *ASGAR*'s learning. To address this challenge, we utilize the Straight-Through Estimator (STE) and Gumbel-Softmax sampling techniques.

### 5.3.2 Improved Variational Component

The variational component of *ASGAR* lacks robustness, mainly because of mode collapse risks and the nature of the loss function. The model tends to be rewarded with favorable loss values even in instances of mode collapse, often because it collapses toward a single strategy that has a high score in all contexts. Consequently, compelling the model to leverage its variational component for diversification, rather than allowing it to collapse, presents a significant challenge.

Given the demonstrated benefits and achievements of vector quantization as a variational technique, we entirely eliminate the conventional variational component in favor of vector quantization. Specifically, we adopt Finite Scalar Quantization (FSQ) [Mentzer et al., 2023] to quantize a context-strategy instance embedding.

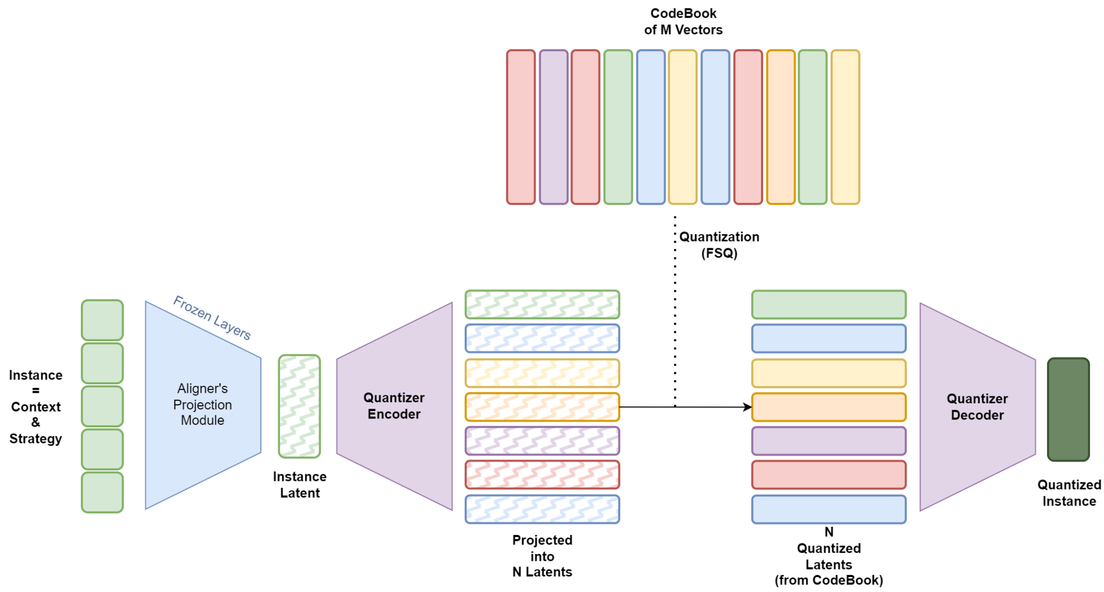



Figure 5.3: Quantization process.

Given a context $c$ and a strategy $s$, we encode this instance using the *Aligner*'s latent representation, projecting it onto a dimensionality of $l_q \in \mathbb{R}^{N_q \times d_q}$, where $N_q$ is the desired number of quantized tokens and $d_q$ is the dimensionality of the latent space in FSQ. Following this, we apply FSQ quantization to transform this projection into $N_q$ tokens, mirroring the approach utilized in VQ-VAE [van den Oord et al., 2017], where each image patch is converted into a number of quantized tokens. These quantized tokens are subsequently passed through a decoder, which reprojects them back to the original dimension of the *Aligner*'s latent representation. Figure 5.3 shows an example of the process for a single strategy.

The aim of this quantization process is to compress the information from the context-strategy instance. To prevent overfitting we need to find an optimal level of compression which holds enough information to differentiate between instances but not enough information to reconstruct an instance. The degree of compression is influenced by the FSQ parameters, the dimensions of the latent space, and the projection layers. This compressed representation (quantized instance) is then fed into *ASGAR*'s decoder alongside

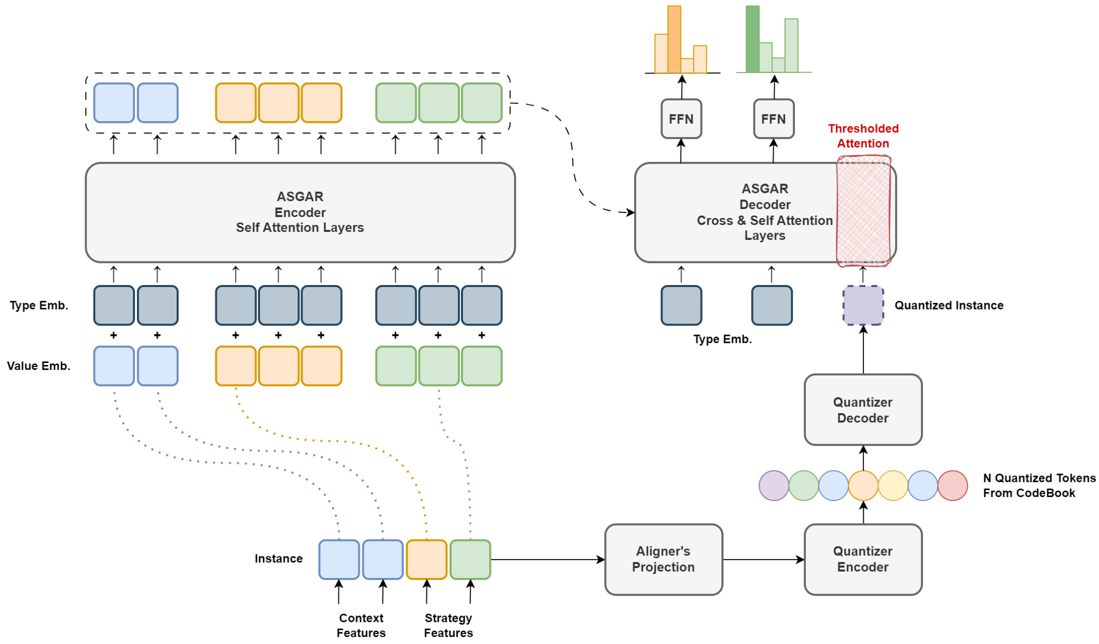



Figure 5.4: ASGAR quantized architecture.

the trigger embeddings. Consequently, while the encoder processes the same context, the decoder receives varying inputs based on the compressed/quantized representation, thereby aiding in the diversification of outputs. The new architecture is shown in Figure 5.4.

To prevent *ASGAR*'s decoder from overly focusing on the quantized representation and overfitting it, we introduce a term in the loss function that reduces the attention weights allocated to this input, confining them within a predetermined threshold $th_{attn-q}$ as shown in Equation 5.4 and Equation 5.5, where $a_q^l$ is the attention weight related to the quantized representation in the $l^{th}$ transformer decoder layer and $\lambda$ is a scaling factor.

$$\mathcal{L}_{attention} = \sum_{l\,=\,1}^{L} \min(a_q^l - th_{attn-q}, 0) \tag{5.4}$$

$$\mathcal{L} = pos.\mathcal{L}_{pos} + avg.\mathcal{L}_{avg} + neg.\mathcal{L}_{neg} + \lambda.\mathcal{L}_{attention} \tag{5.5}$$

This approach encourages the model to employ the representation as a means of diversification, ensuring it conveys just enough information about potential instances without directly leaking this information to its outputs. This strategy works in tandem with adjusting the level of compression to maintain an optimal balance.

During inference, much like VQ-GAN [Esser et al., 2021], in the absence of an instance to quantize, there's a need for a method to generate valid quantization tokens using only the context, without the full strategy instance. To determine the appropriate quantization tokens to feed the FSQ decoder, we adopt a similar approach as outlined in UViM and MaskGIT [Kolesnikov et al., 2022, Chang et al., 2022]. After successfully training $ASGAR$ and achieving satisfactory validation metrics, we can infer that the quantization module is likewise effectively trained. Consequently, we create a new dataset from the original dataset, consisting of rows $(c, q_1, q_2, \ldots, q_N)$, where $c$ represents the context and $q_i$ is the $i^{th}$ quantization token index. We then train a generative model, which could be an autoregressive transformer like GPT or a masked generative transformer akin to BERT. This model $G_q : C \to Q^N$ takes a context $c$ as input and outputs a list of quantized tokens indices, such that it maximizes the likelihood of the quantized indices for that context. as shown in Equation 5.6.

$$\max \prod_{i=1}^{N} P(q_i | c, q_0, \ldots, q_{i-1}) \tag{5.6}$$

In the phase of generating quantized tokens, we can afford to employ sub-optimal sampling methods to broaden the diversity of the outputs. The objective is to produce plausible quantized tokens based on a given context, rather than to identify the optimal set of tokens. The key takeaway here is that tokens which are more probable are closer to those found in the original dataset, whereas the less probable tokens are more exploratory.

After training the quantized token generator, the inference process within $ASGAR$ is updated to the following procedure: for a given context $c$, the token generator is em-

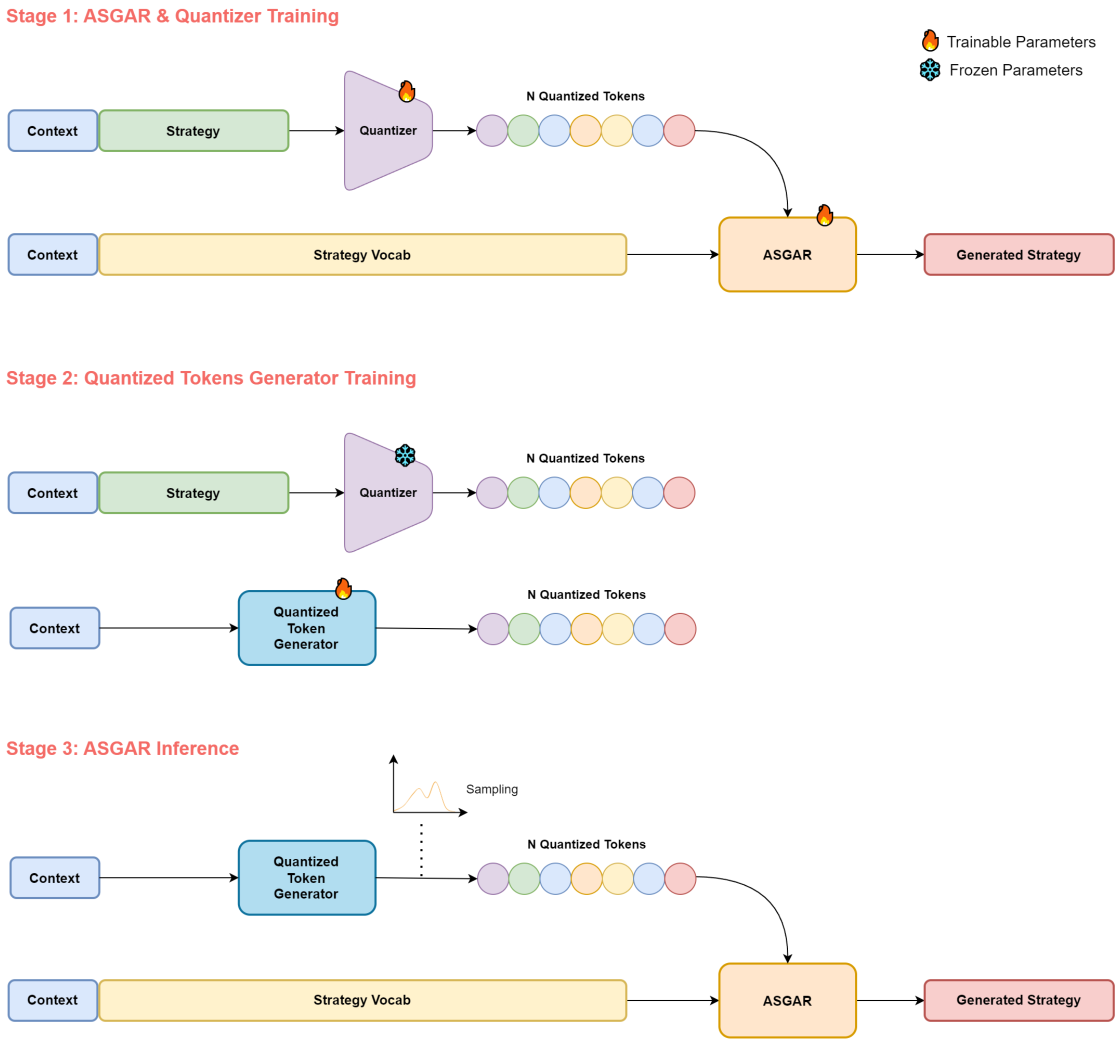



Figure 5.5: Stages of training and inference.

ployed to produce $N$ quantized tokens $(q_1, \ldots, q_N)$. These quantized tokens $(q_1, \ldots, q_N)$, alongside the context $c$ and the strategy features vocabulary, are input into $ASGAR$, which then outputs a strategy $s^G$.

Figure 5.5 shows the overall process of training $ASGAR$, quantized tokens learning, and the final inference process.

This method allows for straightforward manipulation of the quantized tokens, facilitating the easy diversification of $ASGAR$'s outputs. A significant outcome of this flexibility is the ability to decide at inference time whether to generate highly probable quantized tokens, leading $ASGAR$ to produce strategies that are reliable and aligned with the existing data, or to opt for more randomized yet plausible quantized tokens, thereby encouraging $ASGAR$ to explore more novel strategies based on new combinations of quantized tokens while still maintaining a degree of statistical robustness.

## 5.4 Evaluation

### 5.4.1 Enhanced Monitoring & Evaluation Protocol

We propose tracking new, more refined metrics throughout the training process and during evaluation, offering a deeper insight compared to the metrics previously employed.

**Data Fidelity Metrics:** To evaluate the model's ability to accurately replicate positive strategies while steering clear of negative and neutral ones, we introduce the following changes:

- We leverage the *Aligner*'s projected latent representation to measure the Cosine Similarity between generated strategies and the original strategies. On top of employing standard Hamming distance for discrete comparisons.

- Instead of just measuring the overall similarity of the generated strategies with the dataset. Measure three new metrics:
  - **Positive Reconstruction:** measured by the cosine similarity between a generated strategy $\hat{s}$ and its origin strategy $s$. Measured only for positive origin strategies, lower values are desired.

$$R_{pos} = \frac{1}{|S_{pos}|} \sum_{s \in S_{pos}} (Aligner_{cosine-similarity}(s, \hat{s}) - 1)^{2 \cdot \tau} \tag{5.7}$$

  - **Average Reconstruction:** measured by the cosine similarity between a generated strategy $\hat{s}$ and its origin strategy $s$. Measured only for average/neutral origin strategies, medium/high values are desired.

$$R_{avg} = \frac{1}{|S_{avg}|} \sum_{s \in S_{avg}} (Aligner_{cosine-similarity}(s, \hat{s}) - 1)^{2 \cdot \tau} \tag{5.8}$$

  - **Negative Reconstruction:** measured by the cosine similarity between a generated strategy $\hat{s}$ and its origin strategy $s$. Measured only for negative origin strategies, higher values are desired.

$$R_{neg} = \frac{1}{|S_{neg}|} \sum_{s \in S_{neg}} (Aligner_{cosine-similarity}(s, \hat{s}) - 1)^{2 \cdot \tau} \tag{5.9}$$

- We monitor the relationship between overall similarity and the diversity of generated strategies, as indicated by the count of unique outputs. This is crucial because high similarity can still occur in instances of mode collapse, necessitating an examination of similarity in conjunction with generated strategy diversity.

**Generated Positivity:** Rather than assessing and comparing the absolute score values of generated strategies, we focus on evaluating the strategies' positiveness within their specific context, which is more suited to the task at hand and to monitor when convergence plateaus. This approach acknowledges that a strategy scoring 100 in one context might be more desirable than another strategy with the same score in a different context, due to the contextual dependence of score quality. To standardize the comparisons, we adopt the positiveness function $P(c, u)$ as normalization method that quantifies the positiveness of a strategy score $u$ based on the given context $c$.

**Diversity Metrics:** To measure the diversity of generated strategies:

- **Strategy Diversity:** We track the number of distinct generated strategies to measure the diversity on the validation dataset. This allows us to assess diversity early in stage 1, even before the training of the quantized token generator begins.
- **Feature Diversity:** We measure the difference in features within batches to measure feature diversity. This serves as an efficient method to detect potential mode collapses. For instance, while 10 different strategies might appear as distinct and be counted as 10, they could in reality share identical feature values across all their features except one varying feature, which is less desirable. We define the function $Diversity$ which takes as input a batch $B$ of generated strategies as follows:

$$Diversity(B) = \frac{1}{|strategyfeatures|} \sum_{feature \in strategyfeatures} \frac{|Distinct(B_{feature})|}{|B_{feature}|} \tag{5.10}$$

**Inference Time Exploration Effectiveness:** We measure the effectiveness of inference-time exploration via quantized tokens by comparing the Hamming and $Aligner_{cosine-similarity}$ distances between the dataset and $ASGAR$'s outputs. We use the quantized tokens generated by the quantized token generator, initially sampling the most probable tokens. In subsequent iterations, we progressively sample less likely tokens, culminating in the selection of completely random tokens. This method is designed to explore how the likelihood of chosen quantized tokens impacts the performance of $ASGAR$.

**Quantization Instance Leak:** Input leaking happens when instead of learning a generalizable mapping of inputs to quantized vectors, the system might start memorizing specific input patterns. This is more likely when there are too many vectors in the

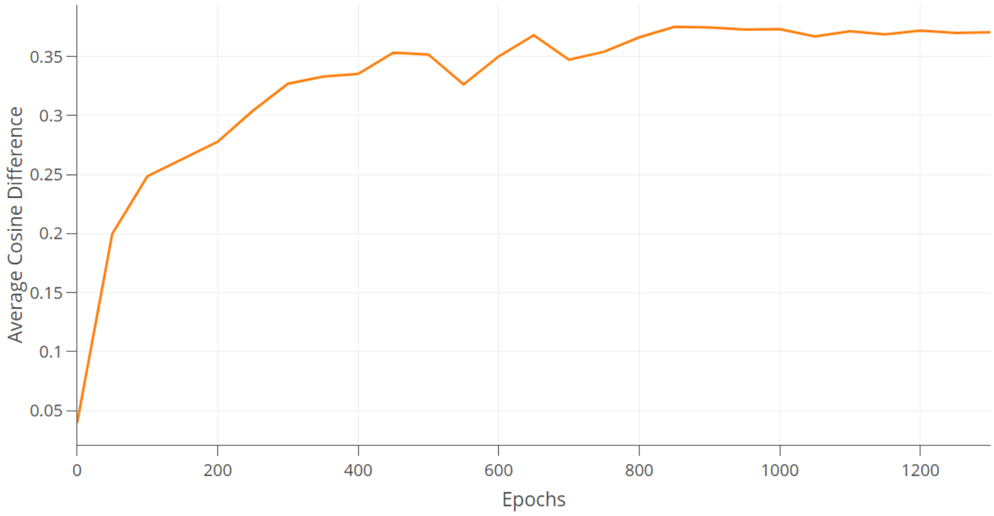



Figure 5.6: Evolution of the difference in cosine similarity between positive and negative examples during training.

codebook relative to the complexity and amount of training data, allowing the model to effectively "remember" input details by mapping them directly to unique vectors and therefore leaks the inputs to the output. To guarantee that the quantization process maintains an optimal level of information compression without input leaking, we monitor the performance of strategies generated by $ASGAR$ when it operates on entirely random tokens. If the quantization fails to adequately compress the instance information, the introduction of randomness will significantly deteriorate the quality of the generated strategies. Conversely, if the quantization effectively condenses the information, then even strategies produced from completely random tokens should remain sufficiently effective. Over-compression is identified by examining diversity; a scenario characterized by mode collapse or markedly low diversity indicates excessive compression.

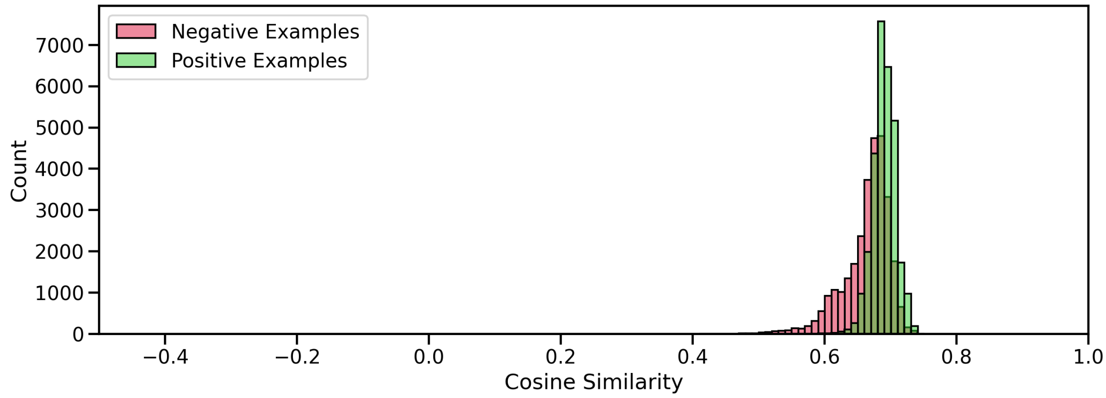
### 5.4.2 Improved Reconstruction

#### 5.4.2.1 *Aligner* Evaluation

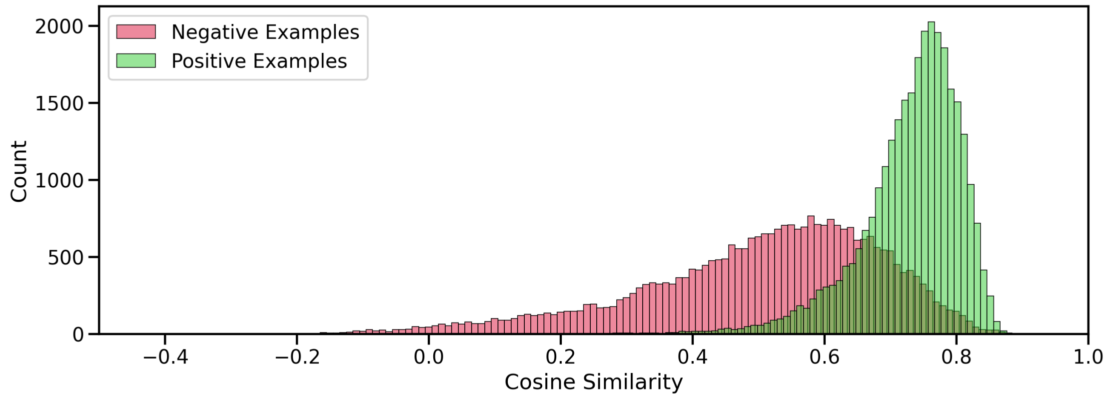
We evaluate the differentiation capability of the *Aligner* by comparing each strategy against all its corresponding positive and negative examples, utilizing the cosine similarity of the *Aligner*'s projected latent representation to gauge separation. We monitor $\Delta_{cosine}(s)$ the average difference in cosine similarity between positive examples $S_s^+$ and negative examples $S_s^-$ of a given strategy $s$ during training epochs, as shown in Equation 5.11. Figure 5.6 shows the evolution of the average difference across all the validation dataset over iterations. We observe that the model clearly increases the gap in cosine similarity between positive and negative examples.

$$\Delta_{cosine}(s) = \frac{1}{\left|S_s^+\right|} \sum_{s^+ \in S_s^+} Aligner_{cos-sim}(s, s^+) - \frac{1}{\left|S_s^-\right|} \sum_{s^- \in S_s^-} Aligner_{cos-sim}(s, s^-) \tag{5.11}$$

Figure 5.7 gives more insight than just cosine similarity difference and shows the distribution of cosine similarity scores for positive examples in green and negative examples in red. We notice that the *Aligner* is able to bring positive examples closer in terms of cosine similarity as the distribution seems more shifted towards 0.8 rather than the starting point of around 0.68. We also notice that negatives samples cosine similarity is more spread out while also being lowered in average. However, there is be some outlier cases where cosine similarity gets lower for positive examples or highed for negative examples, which is the inverse of what we want. Nevertheless, these reverse cases are a clear minority and could be explained by the high level of noise in the training dataset, we still note an overall improvement in terms of strategy separation.




(a) Distribution of Cosine Similarities at the start of *Aligner* training.




(b) Distribution of Cosine Similarities at the end of *Aligner* training.

Figure 5.7: Evolution of the distribution of cosine similarity between strategies and their positive examples (Green) and their negative examples (Red).

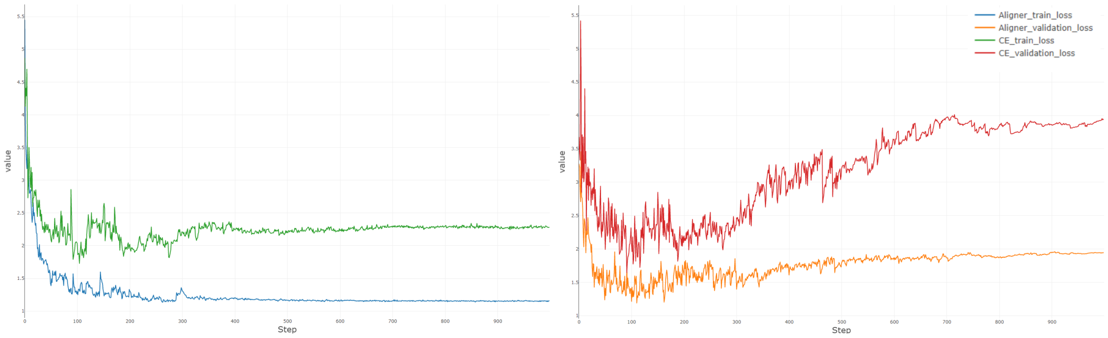
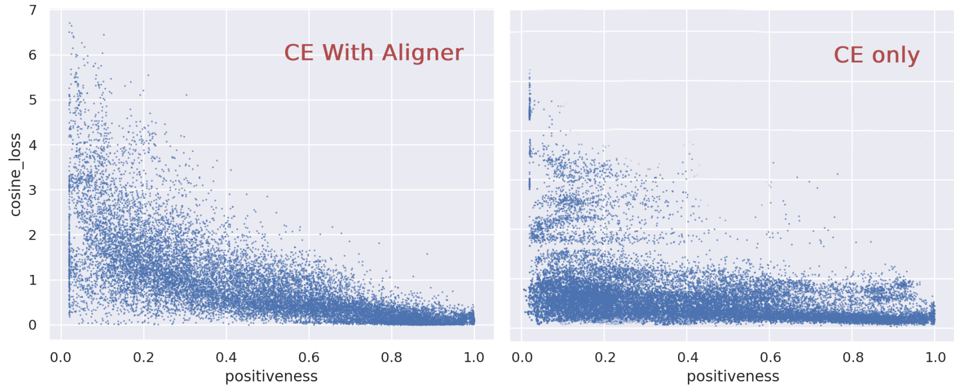



Figure 5.8: The introduction of the *Aligner* to *ASGAR* during training stabilizes the learning and improves the overall loss minimization.

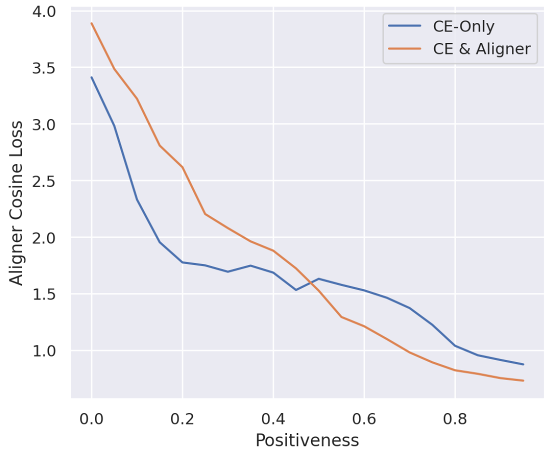
#### 5.4.2.2 Impact on *ASGAR*

To measure the effectiveness of the introduction of the *Aligner* to *ASGAR*, we first check the smoothness of the training loss curve to see if it stabilizes learning by decreasing the confusion related to the old dissimilarity function. As shown in Figure 5.8, we clearly notice a positive impact in terms of training stabilization as the model reaches a lower training and validation loss value overall with less severe fluctuations. We also note that the validation loss doesn't worsen for the *Aligner*-loss case as much as the Cross-Entropy only case, which means that over-fitting happens much sooner in the Cross-Entropy case.

Then, in Figure 5.9, we analyze the similarity (reconstruction loss) of the generated strategies with dataset strategies relative to their positiveness. Note that we use the newly learned *Aligner*'s cosine loss as a reconstruction loss measure in this Figure 5.9 to normalize comparisons between the old method and the newly proposed method. Figure 5.9a shows an example level reconstruction/positiveness scatter plot, on the right we can observe that while using only Cross-Entropy in the loss function there is a somewhat clear "stepping" effect that reflects strategies that are close enough being reproduced in the same way (same reconstruction loss) despite a varying positiveness. However, on the left plot we notice that the addition of the *Aligner* smooths the plot and makes it such that the positiveness of a strategy gradually reflects its reproduction. Figure 5.9b shows an averaged version across all the dataset strategies. We observe that incor-




(a) Cosine Similarity Loss of generated strategies to Dataset strategies based on their positiveness




(b) Average Cosine similarity Loss.

Figure 5.9: Comparison between using Cross-Entropy only as a dissimilarity function VS combining it with the *Aligner*'s Cosine Loss.

porating the *Aligner* leads to a slightly lower reconstruction loss of positive strategies and a significantly better avoidance (or higher reconstruction loss) of neutral and negative strategies. This implies that the model can more effectively distinguish between strategies, even with minor variations in their feature. As a result, it has learned to reproduce more positive strategies and more successfully avoid negative ones, due to receiving clearer and less ambiguous training signals.

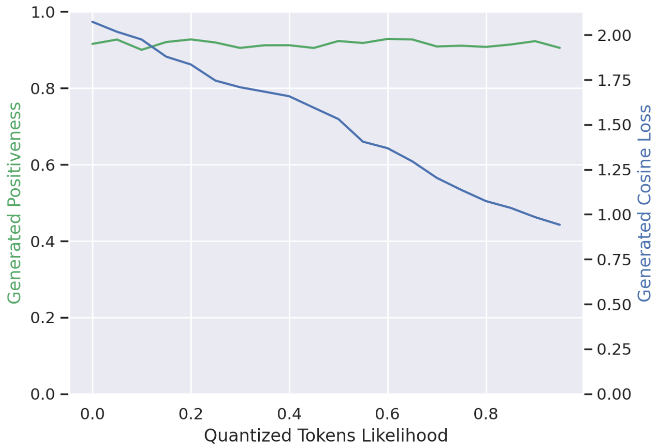



Figure 5.10: The impact of quantized tokens likelihood on the quality of generated strategies by *ASGAR*.

### 5.4.3 Improved Variational Component

We investigate the effectiveness of quantization by comparing the performance of completely random quantized tokens against that of learned quantized tokens. Our findings reveal:

- Figure 5.10 shows that while the performance of totally random tokens falls short of that of learned tokens in terms of data fidelity, it still produces high scoring strategies in terms of positiveness. This suggests that learned tokens introduce a controlled randomness that retains meaningful information.
- The use of random tokens opens up possibilities for a more exploratory mode after the initial training phase, eliminating the need for retraining to achieve higher exploration rates.

The size of the codebook and the number of quantized tokens per strategy are critical parameters that influence the balance between information retention and compression. Table 5.1 shows different parameters and their corresponding impact on the performance of *ASGAR*. These findings indicate:

Table 5.1: Grid search of Quantization compression level.

| **CodeBook size** | **Quantized Tokens** | **Train Loss** | **Val Loss** | **Val Diversity** | **Val Count** | **Random Tokens Positiveness** |
|---|---|---|---|---|---|---|
| 35 | 4 | 1.10 | 1.76 | 0.21 | 57 | 0.92 |
| 35 | 8 | 1.11 | 1.35 | 0.26 | 89 | 0.91 |
| 240 | 8 | 1.02 | 1.27 | 0.31 | 315 | 0.90 |
| 240 | 12 | 0.91 | 1.13 | 0.30 | 1463 | 0.91 |
| 1000 | 16 | 0.82 | 2.41 | 0.27 | 2782 | 0.52 |
| 1000 | 32 | 0.79 | 3.80 | 0.29 | 3929 | 0.43 |

- A large codebook size and a large number of quantized tokens allows excessive information flow, which can cause the model to overfit specific instances, thereby reducing its generalization capabilities. This is shown via a high validation count of generated strategies while yielding low positiveness when given random tokens, which means that the model is heavily relying on the quantized instance that leaks information from the original strategy instance.

- Conversely, a codebook that is too small compresses information excessively, complicating the model's task and potentially leading to mode collapse within specific contexts. This leads to a low validation count despite the good positiveness.

- **Note:** The impact of the codebook size depends on the quantization method that is used. Some quantization methods suffer from low codebook usage, which means that despite a higher sized codebook, the models do not utilize it fully and rely mainly on a few select quantized tokens. That is why, the increase in number of quantized tokens seems to have more impact on our results.

This underscores the importance of selecting an optimal codebook size and number of quantized tokens that ensure efficient information compression without sacrificing the model's ability to generalize.

We observed through our numerous experiments (during development) that using the quantization process greatly outperformed the traditional variational module in promoting diversity. This advantage is especially pronounced in situations with low data fidelity, where the model has the freedom to generate new strategies instead of being constrained to replicate successful strategies from the dataset and therefore has less anchor examples. Previously, the approach was prone to mode collapse if the model's hyperparameters were not properly adjusted or if the model's capacity was insufficient, as shown in Section 4.4.7. In contrast, the quantization method has shown greater resilience to these issues and consistently produced more diverse outputs.

### 5.4.4 Overall Evaluation

In Figures 5.12 and 5.13 we show the monitoring process of data fidelity metrics during *ASGAR* on the validation dataset where we clearly see that the model is gradually learning to reproduce positive strategies, both on the training and validation datasets. Alternatively, we note that the model gradually learns to avoid negative and neutral strategies from the validation dataset. We also note that it avoids neutral strategies less than the negative ones as intended.

Figure 5.11 illustrates that in the initial epochs, the model struggles to diversify its outputs until a notable increase in diversity occurs at a certain point. This delay in diversification is likely attributable to the quantization module's learning curve during the early stages, as it requires time to adequately compress information into the quantized representation to become effective.

Table 5.2 presents a comparative analysis of the previous *ASGAR* model against the current iteration, incorporating the enhancements suggested in this chapter. This comparison utilizes the newly introduced metrics, enabling a more understandable assessment of the models' behaviors. The combination of various metrics is what defines which model is superior. In our case, an improved Positive strategies reconstruction (higher Cosine), an improved feature diversity, an improved Neutral/Negative strategy avoidance (lower Cosine), and a high generated positiveness are aspects that show that our proposed

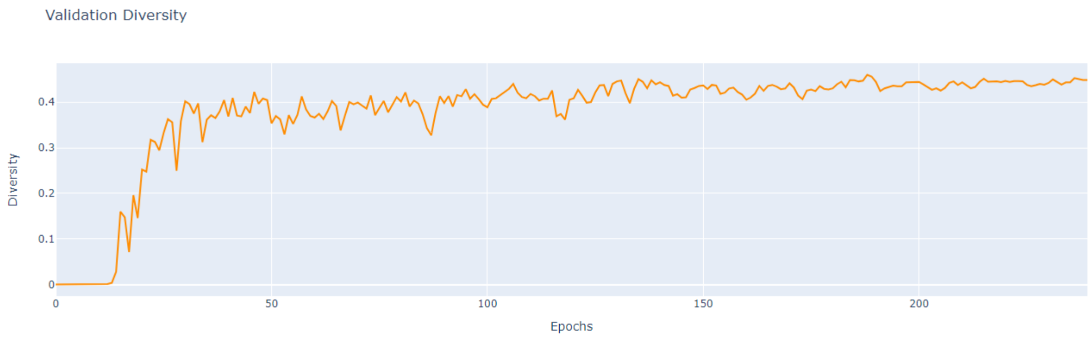



Figure 5.11: Evolution of diversity over epochs.

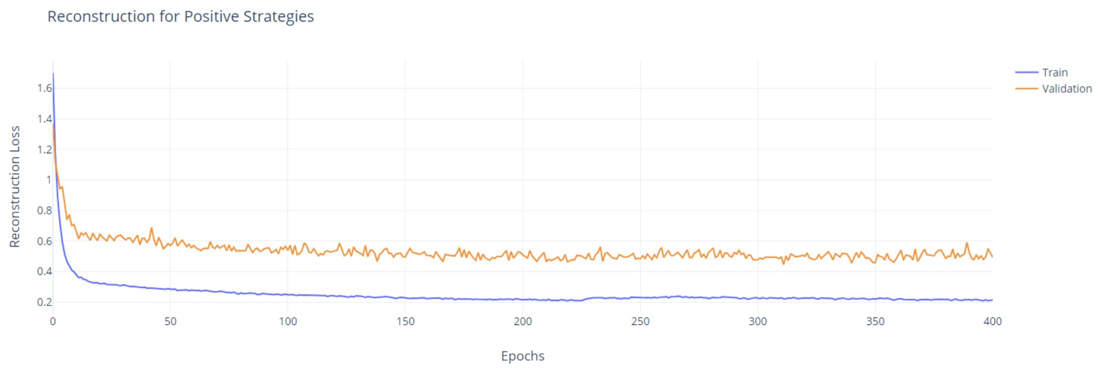



Figure 5.12: Evolution of positive strategies reconstruction over epochs.


Figure 5.13: Evolution of neutral/negative strategies reconstruction over epochs on a logarithmic scale.

Table 5.2: Comparison of the improved current *ASGAR* model vs the previous version.

| Model | Diversity | Positive Cosine | Neutral/Negative Cosine | Generated Positiveness |
|---|---|---|---|---|
| old ASGAR (High Fidelity) | 0.25 | 0.797 | 0.712 | 0.89 |
| old ASGAR (Low Fidelity) | 0.28 | 0.589 | 0.541 | 0.96 |
| **current ASGAR (Normal Mode)** | 0.31 | 0.867 | 0.435 | 0.91 |
| **current ASGAR (Exploratory Mode)** | 0.37 | 0.842 | 0.526 | 0.93 |

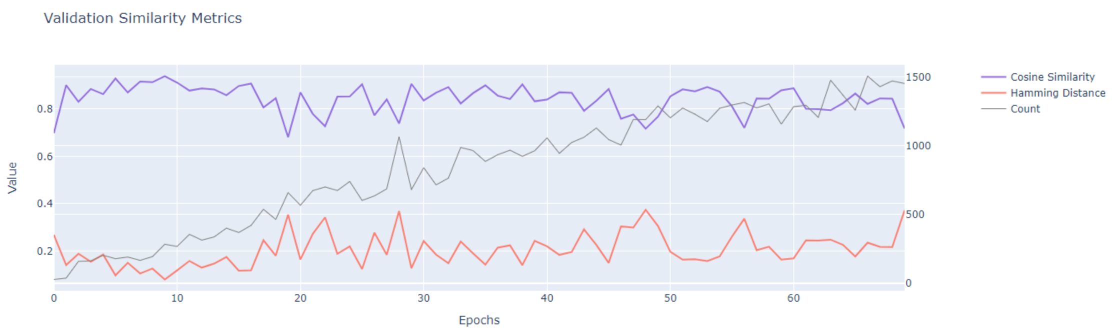



Figure 5.14: Evolution of diversity in relation to similarity over epochs.

enhancement to *ASGAR* have a significant positive impact on its performance.

Figure 5.14 shows the evolution of generated diversity, via the count of unique generated strategies, as well as data fidelity. This indicates that the model starts by generating a few high similarity strategies then slowly learns to diversify its outputs while maintaining high similarity.

### 5.4.5 Discussion

In this section, we discuss the observed advantages as well as the identified limitations of our proposals in this chapter.

#### 5.4.5.1 Advantages

**Smoother Convergence:** The revised approach facilitates a more stable and consistent learning curve, leading to improved model performance and predictability during training. This smoother convergence is beneficial for model development and tuning, as it reduces the likelihood of erratic loss fluctuations and helps in achieving optimal results more efficiently.

**Significantly Reduced Mode Collapse:** By implementing strategies such as quantization, the revised model dramatically lowers the risk of mode collapse, a frequent issue in generative models characterized by overly similar outputs. This improvement provides a more diverse range of outputs, which is crucial to this recommendation/generation task.

**Enhanced Diversity:** The introduction of quantized tokens and an *Aligner*-based approach leads to greater diversity in the model's outputs.

**Inference Time Exploratory Mode Switch:** The updated model includes the capability to toggle between more conservative and more exploratory modes of operation at inference time. This flexibility allows users to adjust the model's output dynamics based on the specific needs of the task at hand, ranging from safer, more predictable outcomes to more novel and diverse options, all without requiring model retraining.

**Enhanced Metrics:** The introduction of new metrics facilitates a deeper and more detailed comprehension of the model's performance. This proves invaluable in the training and evaluation phases, offering a clear basis for comparing the effects of various parameters.

#### 5.4.5.2 Limitations

**Increased Complexity and Dependence on *Aligner* Performance:** The integration of the *Aligner* and the shift towards a quantization-based model architecture introduce additional layers of complexity. The effectiveness of these components is heavily reliant on the *Aligner*'s performance, meaning that any limitations or failures in the *Aligner* could negatively impact the model's overall capabilities and output quality.

**Complexity in Quantized Token Generation Performance:** Similarly, the model's reliance on the performance of quantized token generation adds another layer of complexity. The success of this mechanism is crucial for achieving the desired diversity and reducing mode collapse. However, this necessitates meticulous calibration to maintain an optimal compression level while preventing excessive information loss or leakage, alongside the development of a high-performing quantized token generator.

**Added Paremeters:** Our suggestions result in an increase in hyperparameters that necessitate adjustment, potentially leading to a process that is both resource-intensive and time-consuming.

## 5.5 Conclusion

In this chapter, we presented a series of enhancements to improve the performance and diversity of our previous model (*ASGAR*) as well as a refined evaluation protocol.

A learned metric network (*Aligner*) was introduced to address a limitation in *ASGAR*'s dissimilarity function and ensure sensitivity to minor feature differences. This network's role in distinguishing strategies, reducing confusing signals during training, and boosting overall performance was examined.

We also incorporated a quantization process, effectively overcoming traditional variational module limitations, especially in reducing mode collapse and enhancing output diversity. Crucially, this introduced a novel mechanism for regulated exploration at inference-time, whose effectiveness was evaluated.

Additionally, we developed clearer and more relevant evaluation metrics for a more comprehensive assessment of these improvements and a better understanding of *ASGAR*'s training dynamics. We concluded by reviewing the strengths and limitations of our approaches.

As a result, we observed that our approach has successfully mitigated mode collapse and achieved a notable increase in generative diversity. The exploratory mode during inference time has shown to be effective, delivering excellent results without compromising the overall performance, which is essential for clients interested in exploring new strategies without the necessity of a dedicated model trained in exploratory mode. The enhanced evaluation protocol, with clearer performance metrics, has confirmed the effectiveness of our methods and provided deeper insights to further refine the training process.

Overall, our contributions detailed in this chapter represent a meaningful advancement in digital advertising strategy design, tackling key challenges like diversity and mode collapse and introducing a new mode for inference-time exploration.

# Chapter 6

# Power of Suggestion

## Strategic Feature Manipulation in Transformer-Based Models: A Novel Token-Driven Methodology

### Contents

The content of this chapter is a follow up to Chapter 5. In this chapter, we introduce a novel token-driven approach to offer a nuanced integration of user preferences while balancing suggestions with *ASGAR*'s autonomous decision-making. This includes specialized token embeddings for strategy features, a scheduled token masking strategy for training optimization, and improvements to the loss function to address mode collapse issues and extreme behaviours of the model. We conduct multiple experiments to assess the impact of our approach over the generative process and quality of our model *ASGAR* and discuss its advantages and limitations.

The rest of this chapter is organized as follows: We start with an introduction in Section 6.1. In Section 6.2 we discuss related work. In Section 6.3 we formulate the objective mathematically and explain our proposed approach. In Section 6.4 we discuss our experiments, analyze our results and discuss the advantages and limitations of our approach. Finally we provide a conclusion in Section 6.5.

## 6.1 Introduction

As we advance towards the development of a personalized assistant within the realm of digital advertising strategy design. Our objective is to design a generative model that excels in autonomously crafting high-quality strategies while also integrating user-specified suggestions or desired characteristics into its outputs when possible. The challenge involves striking a balance between conforming to user preferences and leveraging the model's acquired knowledge. Thus, the model can either generate an output that aligns with user preferences or independently diverge from these preferences when it deems them suboptimal, based on its learning. In essence, this approach is similar to seeking advice from a friend who, when presented with specific preferences, either offers guidance aligning with those preferences or, if unable to accommodate them, suggests an alternative they consider beneficial, implying the initial preferences may not be practical.

Flexible control mechanisms have shown great success in fields such as image and text generation [Radford et al., 2019, van den Oord et al., 2017, Park et al., 2019, Choi et al., 2018], largely thanks to the nature of the employed generative techniques. However, as explored in Chapter 4, implementing these methods in the structured creation of advertising strategies presents challenges. This is because of the particular demands and constraints crucial for effective strategy design—such as being utility-driven, generating non-sequential elements, and handling discrete features—which complicate the smooth integration of these approaches without compromising essential aspects of the task.

While our earlier method for advertising strategy design effectively produces high performing strategies, it employs a rigid conditioning approach through contextualization. This method treats the context as a hard condition, training the model to strictly adhere to the context in its generation process.

This work makes three primary contributions to the field of controllable generative modeling, particularly in the context of digital advertising strategy design with *ASGAR*:

- **Innovation in Adaptive Generation:** We introduce a token-driven approach that meticulously incorporates suggestive preferences into the generative process. By integrating specialized token embeddings with strategy feature embeddings, the model learns to prioritize, ignore, or neutrally consider specific strategy attributes during generation.
- **Strategic Training Enhancements:** To optimize model training, we implement a scheduled token masking strategy. This technique enables the model to operate normally under neutral conditions or actively modify its generation based on user-defined preferences through conditioning tokens. Additionally, we refine the Generator model's loss function to mitigate instances of mode collapse, particularly those arising from attribute avoidance.
- **Comprehensive Evaluation:** Through detailed evaluation and analysis, we assess the efficacy of our method, identify its limitations, and propose directions for future improvements.

Our work represents a significant step toward integrating user preferences into generative models for advertising strategies, achieving a delicate balance between maintaining the model's generative autonomy and accommodating user customization.

## 6.2 Related Work

In the rapidly evolving field of controllable generative modeling, significant progress has been made across various domains, including text and image generation. This section delves into the foundational works and recent innovations that inform and inspire our proposed approach in digital advertising strategy design, highlighting contributions in fine-tuning methods, masking techniques, attention mechanism manipulation, token manipulation, and the use of special embeddings.

### 6.2.1 Conditioning and Finetuning Methods

This approach involves modifying the model to accept additional input that specifies desired attributes of the output. For example, in image generation, a model could take a textual description as input to generate an image that matches the description [Nichol et al., 2022, Ramesh et al., 2022, Patashnik et al., 2021]. In text generation, a sentiment label could steer the model to produce content with a specified emotional tone. This approach is readily adaptable to pre-existing generative models [Mirza and Osindero, 2014, Zhang et al., 2021a, Ilias and Askounis, 2023]. For larger models, where comprehensive retraining is prohibitively expensive, certain techniques allow for fine-tuning on new concepts to enable conditioning, providing an efficient means to augment them with controllable capabilities [Dong et al., 2022, Ruiz et al., 2023].

While these techniques are effective in customizing model outputs, their efficient implementation typically requires retraining or substantial alterations to the model's architecture. The application of these methods to advertising strategy design presents significant obstacles, given the necessity to accommodate a wide range of preferences and the requirement for a more comprehensive dataset that specifies optimal strategies along with their associated preferences for effective fine-tuning. This circles back to the initial problem of combinatorial explosion, making their implementation in this context more complex.

### 6.2.2 Masking Methods

The use of masking strategies in training has been pivotal for enhancing model performance. BERT [Devlin et al., 2019] leveraged masked language modeling (MLM) as a pre-training technique that significantly improved the model's understanding of language context and semantics. Similarly, the authors in [Liu et al., 2019] expanded on this with RoBERTa, optimizing the MLM approach to achieve even greater efficiency and accuracy in tasks.

The fundamental concept behind masking strategies involves concealing certain portions of the model's input with a predetermined schedule or methodology, thereby encouraging the model to infer the missing information over time. In our approach, we adapt this concept by focusing on masking signals or guidance instead of the inputs themselves, offering a nuanced variation to the traditional method.

### 6.2.3 Attention Mechanism Manipulation

For models that employ attention mechanisms, such as Transformers [Vaswani et al., 2017], controlling the focus of the attention layers can guide the model to prioritize certain aspects of the input, thereby influencing the characteristics of the output. The works in [Hertz et al., 2023, Tumanyan et al., 2023] manipulate cross-attention or spatial features weights to edit both global and local aspects of the image by changing the text prompt directly, but they tend to preserve the original layout of the source image and fail to handle non-rigid transformations. The methods in [Tewel et al., 2023, Kumari et al., 2023, Cao et al., 2023] delve further by adjusting the Q (query) and K (key) components of the attention mechanism for more profound control.

This approach stands out as a significant research opportunity, albeit fraught with complexities. A significant challenge is the lack of a clear correlation between attention weights and the model's outputs. Meaning that high attention weights on a token do not guarantee the model will select that token as an output; it might simply indicate that the model considers the token's representation in its calculations before deciding to choose a different token. Moreover, many strategies are still navigating the initial stages of analyzing attention flows within models to boost their interpretability. Only a handful of methods that effectively adjust attention signals operate under the assumption that attention weights are directly interpretable and connected to the model's outputs.

### 6.2.4 Token Manipulation

Manipulating quantized tokens to steer the generative process presents an intriguing strategy, especially when connections between specific codebook vectors and particular output characteristics are made. This method, which encodes input data into a discrete set of vectors, effectively acts as a form of conditioning. It allows for precise control over the generated content by determining which codebook vectors (quantized tokens) are introduced to the model to shape the desired outcomes. For instance, by identifying codebook vectors correlated with specific output features, these vectors can be utilized to direct content generation toward those features.

This approach relies on a deep understanding of the model's encoding and quantization processes, as well as the semantic significance of its codebook vectors, in order to provide a precise way to influence generative outputs. It's an area where the intersection of model interpretability and control mechanism design is crucial, as explored in works like [van den Oord et al., 2017,Razavi et al., 2019] or MusicGen [Copet et al., 2023] which leverages quantized audio tokens for melody conditioning. These studies lay foundational insights into the potential of quantized token manipulation in enhancing the specificity and relevance of generative models' outputs.

### 6.2.5 Special Embeddings

The use of specialized embeddings, such as positional embeddings, has significantly contributed to the advancement of generative models. Transformers [Vaswani et al., 2017] introduced positional embeddings to gain the ability to recognize word order, a crucial feature for generating coherent sequences. This innovation has spurred further research into embeddings that encode various types of information, enhancing the model's contextual awareness. BERT [Devlin et al., 2019] utilizes both positional embeddings and segment embeddings to understand the relative positions of words in text and to differentiate between sentences in tasks that involve pairs of sentences. *ASGAR* (our work) and Conna [Benamara and Viennet, 2023, Wei et al., 2022a] use type-embeddings triggers that are added to inputs to dictate to a transformer decoder the type of desired output at each position.

While special embeddings significantly enhance model performance, they can also act as mechanisms for control through strategic manipulation. In our proposed method, we utilize type-embeddings akin to those found in triggers, employing them as a signaling mechanism.

## 6.3 Methodology

This section outlines our proposed method for incorporating user preferences into the generation of advertising strategies, with a particular focus on its implementation within our prior work, *ASGAR* [Benamara and Viennet, 2023]. Initially, we define a user preference and establish the overarching objective of the model. Subsequently, we introduce the novel architecture and training strategy designed to accommodate user preferences. Following this, we suggest a modified loss function to address certain challenges unique to this new approach.

### 6.3.1 Definitions & Formulations

Within the framework of generating advertising strategies, we conceptualize user preference or suggestion as follows: Users may have specific attributes they wish to target for a particular strategy feature (for instance, preferring *"Mobile"* over the model-generated *"Desktop"* for the *"Device"* category). Here, the user effectively "suggests" their preferred attribute to the model. However, they remain open to the model overriding this preference and selecting an alternative attribute for that feature, should it determine that adhering to the user's preference would result in a less effective strategy.

Let $S = \{s_1, s_2, \ldots, s_m\}$ be the set of strategies, where each strategy $s_i$ is defined as a combination of feature categories. Let's assume there are $N$ feature categories in total, denoted by $F = \{f_1, f_2, \ldots, f_N\}$.

Each feature category $f_i$ has its own subset of attributes $A_i = \{a_{i1}, a_{i2}, \ldots, a_{ik_i}\}$, where $k_i$ represents the number of attributes in the subset corresponding to the $i$-th feature category.

Therefore, a strategy $s_i$ can be represented as a vector of selected attributes from these subsets, such that:

$$s_i = (a_{1x}, a_{2y}, \ldots, a_{Nz})$$

where $a_{1x} \in A_1, a_{2y} \in A_2, \ldots, a_{Nz} \in A_N$ are the selected attributes for the $i$-th strategy across the $N$ feature categories.

Let $P_i$ represent the set of user preferences, where each preference for a feature category $f_i$ includes multiple attributes from its subset $A_i$. We can then define a user preference as $P_i = \{(f_i, A'_i)\}$, where:

- $f_i$ is a specific feature category from the set of feature categories $F$.
- $A'_i \subseteq A_i$ represents the set of preferred attributes within the subset of attributes $A_i = \{a_{i1}, a_{i2}, \ldots, a_{ik_i}\}$ for the feature category $f_i$. Here, $A'_i$ can contain one or more attributes, indicating multiple preferences within the same category.

Thus, if a user prefers multiple attributes $a_{ix}, a_{iy}, \ldots$ within the same feature category $f_i$, the user preference is represented as:

$$P_i = \{(f_i, \{a_{ix}, a_{iy}, \ldots\})\}$$

Therefore, Let $P$ be the set representing the complete set of user preferences, where each element of $P$ is itself a set of preferences related to a specific feature category. Formally, we define $P$ as:

$$P = \{P_1, P_2, \ldots, P_N\}$$

Given a strategy generation model $G$, which takes as input a context $c$, the goal of $G$ is to output a strategy $s$ that maximizes the estimated score $E(c, s)$ in context $c$, while also ensuring the strategy's likelihood with respect to a training dataset $D$, and incorporating user preferences $P$. The objective for $G$ can be expressed as:

$$s^* = \underset{s}{\operatorname{argmax}} \left( E(c, s) + \lambda_1 \mathcal{L}((c, s); D) + \lambda_2 R(s, P) \right) \tag{6.1}$$

Where:

- $E(c, s)$ denotes the estimated score of strategy $s$ given context $c$, with the model aiming to maximize this score.
- $\mathcal{L}((c, s); D)$ measures the likelihood of strategy $s$ with context $c$ being similar to those in the dataset $D$, ensuring the generated strategy is grounded in realistic, data-driven examples.
- $R(s, P)$ quantifies how well the strategy $s$ aligns with the user preferences $P$, allowing for the generation of strategies that are tailored to user specifications.
- $\lambda_1$ and $\lambda_2$ are weighting parameters that respectively balance the importance of the strategy's data fidelity and adherence to user preferences, both in relation to score maximization.

This definition of $G$ presents a comprehensive goal for the model: to create strategies that are effective (as shown by $E(c, s)$), reflective of real-world strategies (as informed by $\mathcal{L}((c, s); D)$), and tailored to user preferences (as directed by $R(s, P)$). This strategy ensures a refined equilibrium among high performance, data fidelity, and personalization in the advertising strategies it generates.

We can define $R(s, P)$ as a function that quantifies the adherence of the generated strategy $s$ to the set of user preferences $P$. Recall that $P_i = \{(f_i, A'_i)\}$, where $f_i$ denotes a feature category and $A'_i$ represents the subset of preferred attributes within that category specified by the user. Given a strategy $s$ represented as a combination of selected attributes for each feature category, $s = (a_{1x}, a_{2y}, \ldots, a_{Nz})$, the compatibility function $R(s, P)$ can be defined as follows:

$$R(s, P) = \frac{1}{|P|} \sum_{(f_i, A'_i) = P_i \in P} \delta(a_i^s, A'_i) \tag{6.2}$$

Where:

- $a_i^s$ is the attribute selected in strategy $s$ for the feature category $f_i$.
- $\delta(a_i^s, A'_i)$ is a function that returns a positive value if $a_i^s \in A'_i$, indicating that the selected attribute $a_i^s$ for feature category $f_i$ in strategy $s$ is one of the preferred attributes specified by the user in $A'_i$. Otherwise, it returns a lower value or zero, indicating non-adherence.

The function $\delta$ can be designed to assign higher scores for exact matches between the strategy's attributes and the user's preferred attributes, thereby quantifying the extent to which $s$ respects the user's preferences $P$ across all specified feature categories. The sum over all user-specified preferences ensures that the overall measure $R(s, P)$ reflects the cumulative adherence of the generated strategy to the entire set of user preferences.

This formulation allows $R(s, P)$ to serve as a comprehensive metric for evaluating the alignment of the generated strategy with user preferences, facilitating a balance between generating high-scoring, realistic strategies and tailoring the output to meet user-specified preferences.

### 6.3.2 Special Token-Driven Learning

Building on the definitions outlined in the prior section, integrating user preferences into the model for generating advertising strategies necessitates the addition of a term to optimize $R(s, P)$. This function measures how closely a generated strategy $s$ aligns with a set of user preferences $P$. A challenge arises, however, due to the nature of the data, which typically do not pre-include user preferences $P$. Consequently, it becomes necessary to simulate user preferences during the training process. Unfortunately, this reintroduces the issue of combinatorial explosion, as it would require iterating through all possible preferences to adequately train the model on responding to the presence of user preferences.

Fortunately, the design of our previous model, *ASGAR*, enables us to integrate user preferences without resorting to combinatorial iteration over those preferences. To facilitate the model's accommodation of user preferences $P$ during the generation process, we suggest exploiting *ASGAR*'s capability to process the entire vocabulary of strategy features. By treating a subset of feature attributes as user preferences, we can enhance the dataset examples with these inferred preferences. We suggest dynamically sampling these user preferences based on a parameterized schedule.

We introduce a selection function, denoted as *select*, which is dynamically applied to all strategies within a batch during training. This function selects a subset of attributes from the original strategy based on the selection ratio $r$, which determines the proportion of attributes to be included in the inferred preference set $P'$. The selection function is activated for each dataset example during training, incorporating a randomization process that guarantees variability in selection across different batches, thereby ensuring maximum coverage of possible preference sets. As illustrated in Figure 6.1, we define the process as follows:

1. **Define the Strategy:** Consider a strategy $s$ that consists of $N$ feature attributes

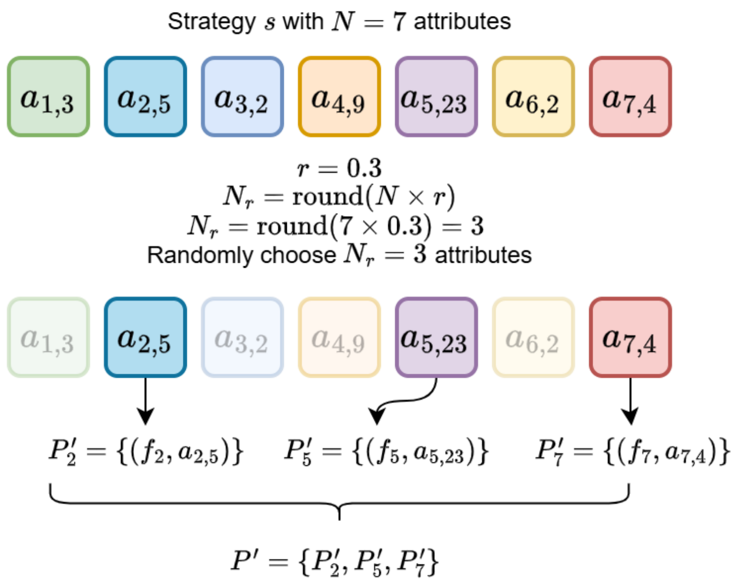



Figure 6.1: The process of preference extraction from the dataset examples.

represented as $s = (a_{1x}, a_{2y}, \ldots, a_{Nz})$, where each $a_{iy}$ corresponds to an attribute in the $i$-th feature category.

2. **Determine the Number of Attributes to Select:** Calculate $N_r$, the number of attributes to select, by rounding $N \times r$ to the nearest integer. $N_r = \text{round}(N \times r)$.

3. **Randomly Select Attributes:** From the total $N$ feature attributes, randomly choose $N_r$ unique indices from the set $\{1, 2, \ldots, N\}$. Denote this subset of indices as $J$.

4. **Construct Preferences Set:** For each index $j \in J$, select the attribute $a_{jy}$ in $s$. This results in a subset of strategy attributes $s' = \{a_{jy} \mid j \in J\}$.

   The selection function, which generates the new set of attributes based on a random sampling ratio $r$, can be mathematically described as:

$$s' = \text{select}(s, r) = \{a_{jy} \mid j \in J\}$$

Using this selection function, we can build the set of inferred preferences $P' =$

$\{P'_j \mid j \in J\}$ such that $P'_j = \{(f_j, a_{jy})\}$.

This method facilitates the simulation of user preferences in a manner that ensures these preferences are consistent with the strategies found within the dataset.

Given that *ASGAR* processes the entire vocabulary of strategy features through its transformer encoder, we introduce a token-driven approach for soft conditioning. Inspired by the type-embedding triggers in its decoder, our proposed method involves augmenting each feature attribute embedding with a special embedding before inputting it to the encoder. This additional embedding is designed to signal to *ASGAR* the intended treatment of a particular feature attribute in the generation process. A straightforward implementation of this concept would involve the creation of three distinct special embeddings: *"Favor"*, *"Avoid"*, and *"Neutral"*. These Special Tokens are instrumental in guiding the model towards generating strategies because:

- ***Favor*:** indicates that the model should prioritize outputting this feature attribute. During training, when an original strategy $s_i$ is positive (based on the positiveness function), all of its selected attributes using the *select* function during the selection process are combined with the embedding of the *"Favor"* token.
- ***Avoid*** indicates that the model should prioritize avoiding this feature attribute. During training, when an original strategy $s_i$ is negative (based on the positiveness function), all of its selected attributes using the *select* function during the selection process are combined with the embedding of the *"Avoid"* token.
- ***Neutral*:** indicates that the model should just operate normally (this is equivalent to not combining any embedding). In this case, all of the inputs to *ASGAR* are combined with the *"Neutral"* embedding.
- It's important to highlight that any feature attribute not chosen in the selection process is automatically paired with the *"Neutral"* embedding. This default action


Figure 6.2: *ASGAR* input construction with special tokens.

guarantees that the model retains the ability to operate independently, making decisions without explicit guidance.

These embeddings are learned during training and remain consistent across all instances. For example, there is a singular *"Favor"* embedding that is applied (or combined) wherever necessary.

Figure 6.2 shows how the new input of special tokens is build and prepared to be combined with the base vocabulary input of *ASGAR*. Note that some of the tokens will be converted back to a *"Neutral"* token due to the preference extraction rate $r$ to apply the masking schedule and avoid input leaking.

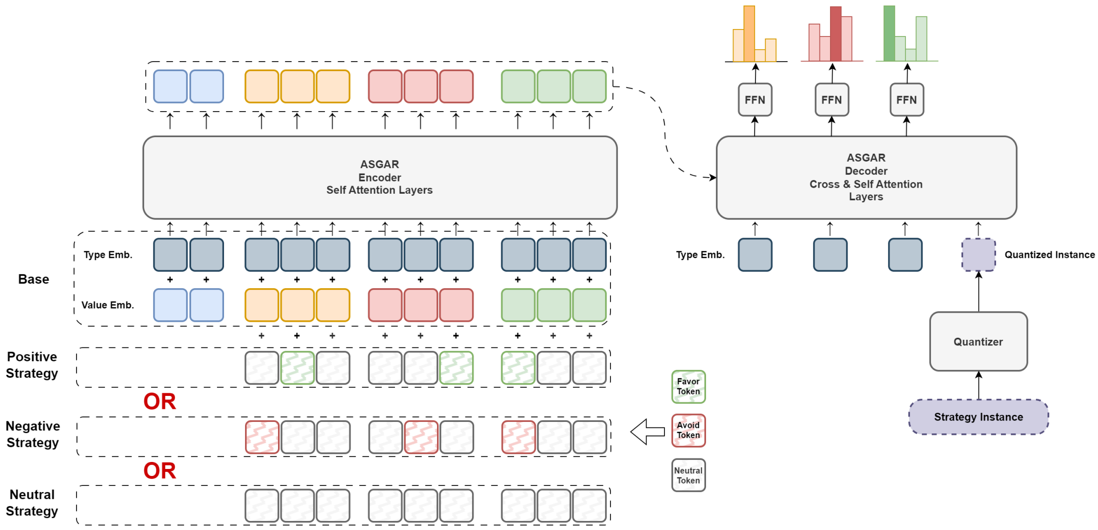



Figure 6.3: Special Token-Driven *ASGAR* Framework.

Figure 6.3 illustrates the overall architecture we propose with an example depicting the incorporation of special tokens in a batch of strategies.

For training purposes, we propose a selection scheduling approach that initiates with a maximal selection ratio $r_{max}$, allowing *ASGAR* to start learning with heavy token guidance. Then, we gradually decrease this ratio through successive iterations until it attains a predetermined minimum selection ratio $r_{min}$. This scheduling strategy begins with strong token guidance, where the model learns to respond to various tokens. It's important to note that a 100 percent selection ratio can lead to input leaking because it directly instructs the model on which attributes are important to minimize loss. Gradually, we reduce the visibility of these tokens, shifting towards predominantly *"Neutral"* tokens. This transition helps the model revert to its standard mode, effectively transferring the knowledge of strategy generation to the *"Neutral"* tokens while maintaining the influence of *"Favor"* and *"Avoid"* tokens when provided.

### 6.3.3 Avoiding Extreme Behaviour

The introduction of the *"Favor"* token effectively complements and aligns with the original goals of generating high scoring strategies with high data fidelity. However, the "Avoid" token presents a unique challenge by potentially driving the model to overly conform to preference avoidance. This could result in the model optimizing adherence to these negative preferences to such an extent that when given an *"Avoid"* token it converges all its outputs towards a singular, high-performing strategy that is effectively dissimilar from all negative strategies from the dataset (which still minimizes the loss function). Consequently, this behavior could initiate in a new form of mode collapse, triggered specifically by the model's response to an *"Avoid"* token.

The issue primarily stems from how *ASGAR*'s loss function is structured:

- For positive strategies, the loss decreases as the dissimilarity between the original and generated strategies diminishes.
- For negative strategies, the loss is reduced by both maximizing the dissimilarity between the original and generated strategies and ensuring the generated strategy has a high estimated score. This is exactly where the problem occurs. The model here might converge towards a strategy that is both highly dissimilar from the negative examples and exhibits strong performance, leading to mode collapse. This issue was less apparent in previous works, as the primary focus was on enhancing the reconstruction of positive strategies, ensuring that negative ones are effectively bypassed.

We suggest revising the components of the original loss function related to strategy avoidance with the following rationale:

- Rather than merely aiming to increase dissimilarity with negative strategies as a way to avoid them — a process that risks mode collapse by potentially driving the model towards a singular strategy — we propose a more controlled approach that offers additional guidance to refine this strategy.

- Rather than maximizing Cross Entropy Loss to increase dissimilarity with negative strategies, we propose reversing the approach by minimizing Cross Entropy Loss for all feature attributes except those we aim to avoid. This strategy is designed to steer the model away from specific attributes while encouraging it to engage with all other attributes in a more evenly distributed manner (based on its already learned positives distribution), which penalizes the model if it disproportionately favors any single attribute.
- An intuitive method to achieve this would be to lower the probability of the "bad" attributes and proportionally increase the probability of other attributes, as shown in Equations 6.3 and 6.4.
- To ensure that the training of positive examples remains undisturbed (as manipulating output probabilities could potentially impede learning), we initially allow the model to train for several iterations without this change in objective. This establishes baseline distributions that indicate the model is beginning to learn to produce useful outputs. After this phase, we transition from the original loss function to the new one. This shift not only maintains the distributions learned during the initial training phase but also subtly adjusts them to account for the avoidance of certain attributes.

Consider a strategy feature category $f_i$ with $k_i$ possible attributes, and let $P(a_{ij})$ represent the probability to output the $j$-th attribute in strategy feature $f_i$. If attribute $a_{ij}$ is identified as an avoided attribute (is in a negative strategy), the process to adjust probabilities is defined as follows:

1. Decrease the probability of the avoided attribute $a_{ij}$, $P(a_j)$, by a certain amount $\Delta$.

2. Equally distribute the amount $\Delta$ among the probabilities of the remaining $k_i - 1$ attributes.

Mathematically, this adjustment can be represented as:

- For the avoided attribute $a_{ij}$:

$$P'(a_{ij}) = P(a_{ij}) - \Delta \tag{6.3}$$

- For each other attribute $o \neq j$:

$$P'(a_{io}) = P(a_{io}) + \frac{\Delta}{k_i - 1} \tag{6.4}$$

Where $P(a_{ij})$ is the original probability of attribute $i = a_{ij}$, $P'(a_{ij})$ is the adjusted probability of attribute $a_{ij}$, $\Delta$ is the reduction in probability for the avoided attribute, parameterized based on the wanted severity of avoidance.

Figure 6.4 illustrates the desired behaviour versus the old behaviour. The goal is to prevent the model from overriding learned distributions to maximize dissimilarity (minimizing loss). Instead, the objective is to retain these learned distributions while integrating the knowledge that certain attributes are less desirable.

Practically, since we were previously using Cross-Entropy loss as a base dissimilarity function. We make two changes to implement the needed behaviour:

- We adjust the target probability distribution used in the Cross-Entropy loss calculation for negative strategies to represent $P'(a_{ij})$.
- The objective shifts towards minimizing this modified Cross-Entropy loss term. Consequently, we substitute $\mathcal{L}_{neg}$ with a function similar to $\mathcal{L}_{pos}$ (the loss terms from *ASGAR* in Chapter 4), albeit employing the revised Cross-Entropy loss term.

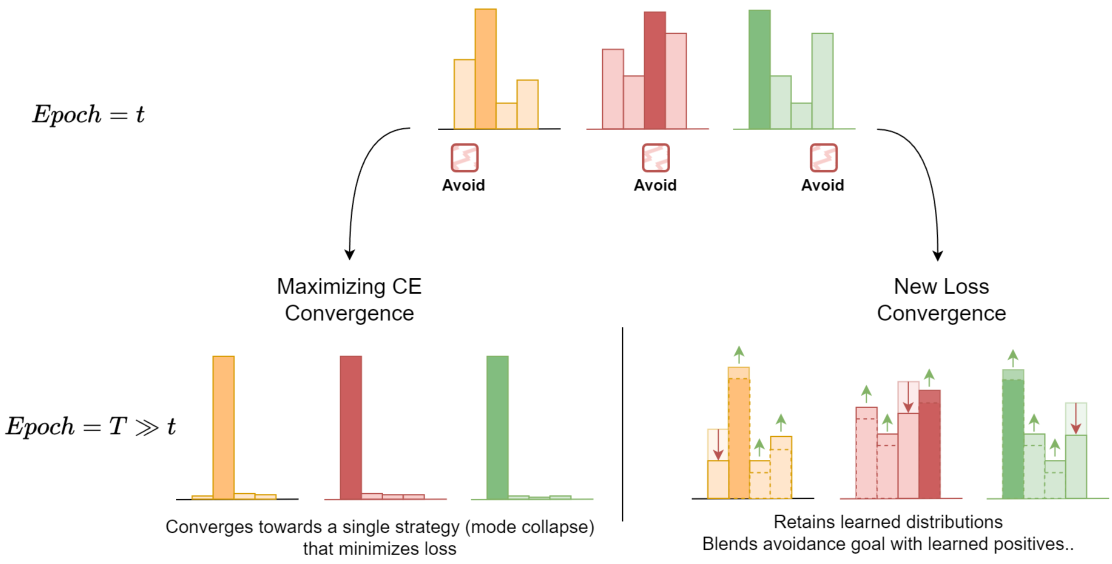



Figure 6.4: Illustration of feature avoidance probability redistribution.

## 6.4 Experiments & Results

### 6.4.1 Evaluation Protocol

We conduct a comprehensive analysis to assess the impact of incorporating user preference tokens into the *ASGAR* model. Our evaluation strategy is methodically structured around several key experiments, each designed to examine the model's performance under varying conditions and to analyse the effectiveness of the token-based guidance mechanism. The following areas form the core of our evaluation:

- **Standard Generation Process Evaluation:** measure any potential degradation in performance resulting from the introduction of preference tokens.
- **Mode Collapse Investigation with *"Avoid"* Tokens:** detecting the emergence of mode collapse specifically when *"Avoid"* tokens are employed.
- **Effectiveness of Token Guidance:** To examine the influence of token guidance on strategy generation, we propose a multi-level analysis:

  - **Single Token Suggestion** We explore the impact of applying a singular *"Favor"* or *"Avoid"* token on a specific feature attribute.
  - **Multiple Tokens Suggestion** Similarly, when suggesting multiple attributes, we perform this analysis in two case:
    * Single feature: All tokens pertain to the same feature category.
    * Multiple features: Tokens are distributed across various feature categories.

- **Monitoring of Metrics for Each Experiment:** For each experimental scenario outlined above, we track standard metrics to gauge the generated strategies positiveness and data fidelity. As well as newly defined membership metrics for preference adherence.

Through this structured evaluation, we aim to shed light on the nuanced dynamics of preference-driven strategy generation, exploring the effectiveness and potential limitations of our token-driven approach within the *ASGAR* model.

### 6.4.2 Stability Analysis

Initially, we benchmark the model's performance by utilizing only *"Neutral"* tokens at inference, mirroring the original model's behavior with no preferences. This serves as a control scenario, allowing us to measure any potential degradation in performance resulting from the introduction of preference tokens. Key metrics to monitor include generated strategy positiveness, diversity, and data fidelity.

In Table 6.1, we compare validation metrics of our proposed Token-Driven approach at neutral mode versus the previous performance of *ASGAR* to measure its impact and potential performance deterioration. We analyse both high fidelity and low fidelity modes via only the base model's 6 fidelity parameters as described in Chapter 4. We observe

Table 6.1: Comparison of Token-Driven *ASGAR* model vs the previous version Base ASGAR.

| **Model** | **Positiveness** | | | **Score** | **Aligner Loss** | | | **Ham** | **Cos** | $\lvert S^G \rvert$ | **Div** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | + | ~ | - | | + | ~ | - | | | | |
| old ASGAR (high fidelity) | 0.7775 | 0.8878 | 0.9200 | 122.94 | 0.2810 | 0.7763 | 2.1944 | 0.3648 | 0.7884 | 1072 | 0.4347 |
| old ASGAR (low fidelity) | 0.9187 | 0.9380 | 0.9430 | 137.98 | 0.5425 | 0.9988 | 2.1063 | 0.4444 | 0.7005 | 266 | 0.3912 |
| current ASGAR (high fidelity) | 0.7841 | 0.9112 | 0.9433 | 129.17 | 0.2517 | 1.4066 | 2.7242 | 0.3658 | 0.7639 | 2355 | 0.4409 |
| current ASGAR (low fidelity) | 0.9147 | 0.9498 | 0.9450 | 136.53 | 0.5843 | 1.3778 | 2.7027 | 0.5255 | 0.6195 | 461 | 0.3465 |

that incorporating the guiding tokens mechanism into *ASGAR* does not compromise performance; rather, it enhances certain metrics. Notably, there is an increase in the count of unique generated strategies, improved avoidance of negative and neutral strategies, and a slight enhancement in the reconstruction of positive strategies. This improvement is likely because the signaling tokens offer additional reference points during training, aiding the model in adhering more closely to positive strategies. Consequently, a more informed quantization process promotes greater diversification, leading to a higher number of generated strategies.

A critical aspect of our analysis focuses on detecting the emergence of mode collapse specifically when *"Avoid"* tokens are employed. This is assessed by tracking the count of distinct strategies produced in response to an *"Avoid"* token throughout the entire validation dataset. Figure 6.5 illustrates the evolution of the number of distinct strategies generated by the model when encountering *"Avoid"* tokens, highlighting a progressive increase in their number while still generating high-positiveness strategies. This indicates that the model successfully avoids becoming fixated on a singular strategy, instead learning to broaden its range of outputs.

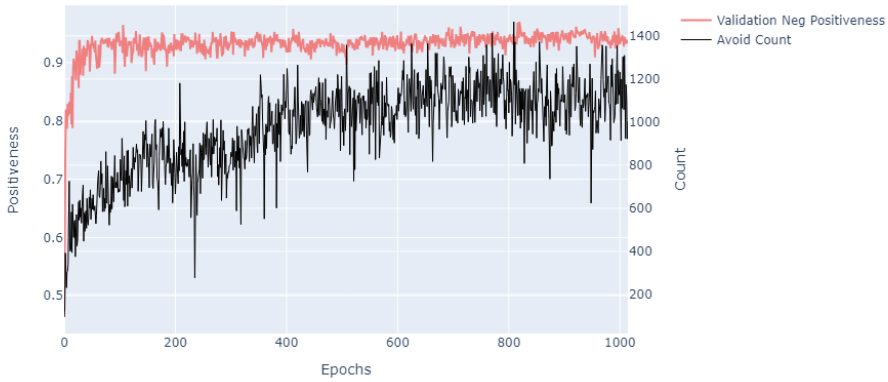



Figure 6.5: Evolution of distinct generated strategy count throughout training for negative strategies in the validation dataset.

### 6.4.3 Effectiveness of Token Guidance

To examine the influence of token guidance on strategy generation, we propose a multi-faceted approach. Our analysis begins with examining the effects of applying a single token suggestion to a singular attribute within a single feature category. Then we examine multi token suggestion in both mono-feature and multi-feature settings.

#### 6.4.3.1 Single Token Analysis

In the single feature single token case, a single *"Favor"* token is used at a time and all the other vocabulary gets a *"Neutral"* token. For every possible context $c \in C$ from the dataset, such that $|C| = 26$, we generate $n = 200$ strategies using entirely random quantized tokens to activate $ASGAR$'s exploratory mode. This process results in a total of 5200 strategies. Our analysis focuses on identifying the number of these strategies that qualify as positive (high performance strategies), defined by achieving a minimum positiveness of $p_{min}$, and subsequently, we determine how many of these positively evaluated strategies incorporate the specifically suggested attribute.

We also measure the uplift in attribute appearance to analyse the effectiveness of our method to output certain attributes more often when they are suggested. The uplift metric is defined as follows:

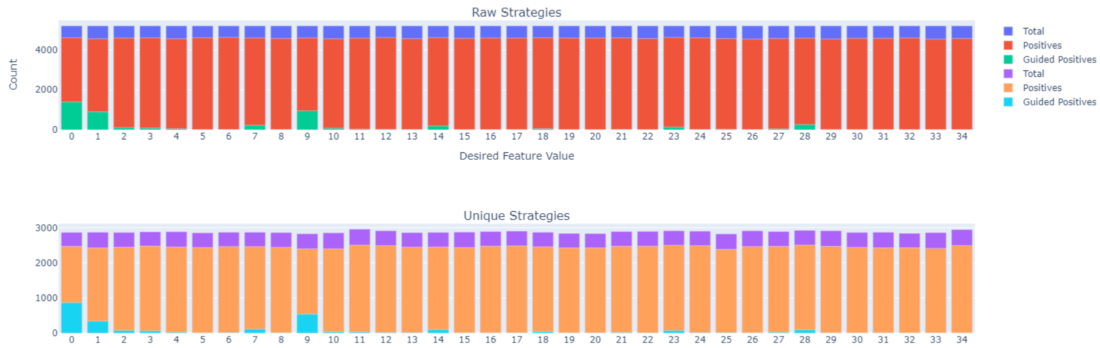
$$U(a) = \frac{\sum_{s^{+} \in S^{favor} | p(c,s^{+}) \geqslant p_{min}} \mathbf{1}_{s^{+}}(a)}{\sum_{s^{\sim} \in S^{neutral} | p(c,s^{\sim}) \geqslant p_{min}} \mathbf{1}_{s^{\sim}}(a)} \tag{6.5}$$

Where $a$ is the suggested attribute, $S^{neutral}$ is the set of generated strategies in neutral mode with no preferences, $S^{favor}$ is the set of generated strategies when putting a "$Favor$" token along with the suggested attribute $a$, and $\mathbf{1}_s$ is the indicator function. In all our experiments, we visually present this uplift through bar charts to simplify the analysis and to prevent depicting infinite uplift in scenarios where an attribute did not previously exist but now does.

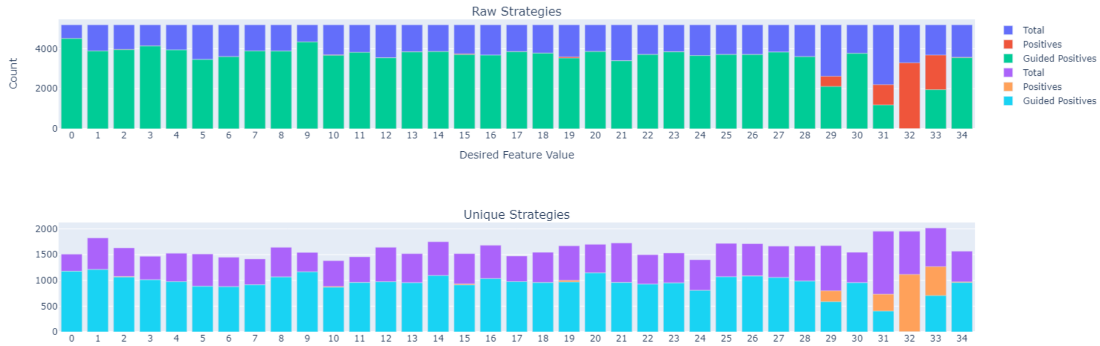
Figure 6.6 displays the effects of employing a *"Favor"* token across different attributes of the *"Region"* feature. It is evident that the introduction of the *"Favor"* token increases the occurrence of positive strategies featuring the recommended attribute in the majority of instances. In situations where the token's influence appears minimal, this can often be traced back to the relative scarcity of those attributes within positive strategies in the training data, as shown in Figure 6.7, suggesting a likely reason for the model's inability to effectively integrate these suggestions.

Likewise, Figure 6.8 and Figure 6.9 reveal a comparable pattern, albeit with a notable distinction: certain attributes, despite their rarity in the training dataset, are still generated more frequently than in standard mode. This suggests that the model is able to interpolate relationships. Additionally, it's observed that the model can generate positive strategies featuring attributes not present in the training dataset (as for attribute *"8"*, affirming its capability to explore and generate novel strategies.

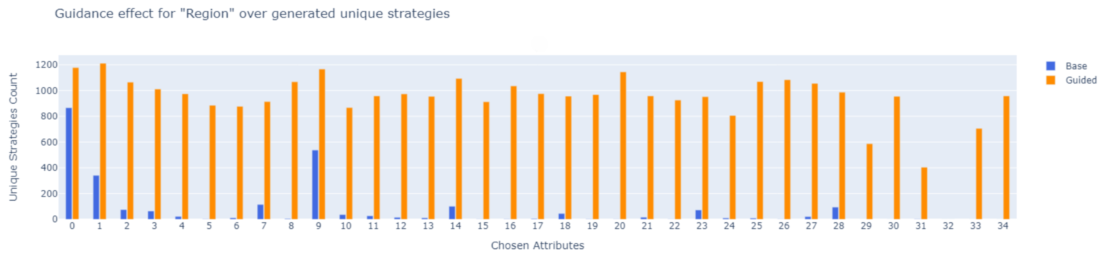
Similarly, we analyze the effect of the *"Avoid"* token. Figure 6.10 illustrates that the model's ability to decrease the presence of an attribute depends upon the existence of sufficient negative strategies containing it in the training dataset. It also hints at the model falling back to the most promising attributes (based on its learning) when asked to




(a) Occurrences of each attribute when generating in standard mode




(b) Occurrences of each attribute when suggesting it during generation.




(c) Comparison of unique strategies count that contain suggested token: Guided vs Base (non-guided).

Figure 6.6: The impact of a *"Favor"* suggestion token over each attribute in the feature category *"Region"*.

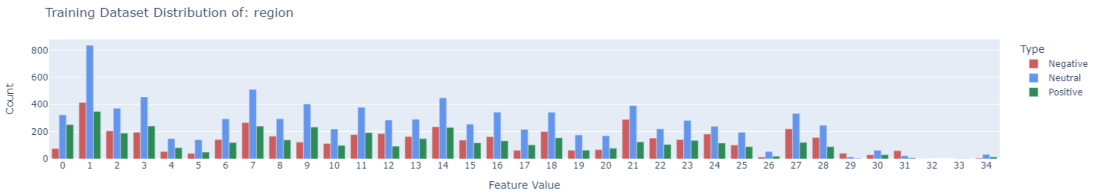
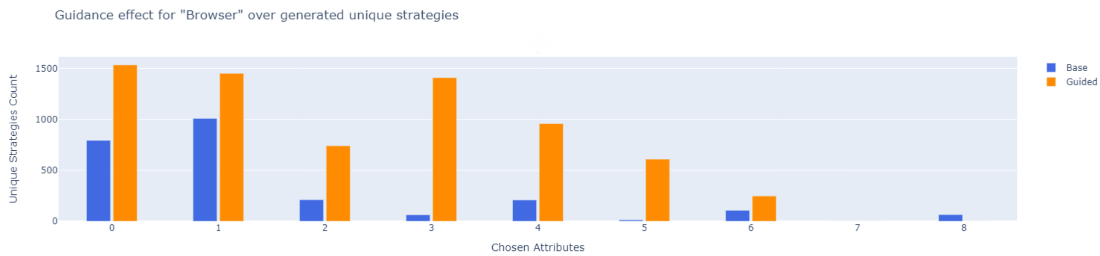



Figure 6.7: Training dataset attribute occurrences of feature category *"Region"* categorized by the positiveness of the strategy they occur in.


(a) Occurrences of each attribute when generating in standard mode


(b) Occurrences of each attribute when suggesting it during generation.




(c) Comparison of unique strategies count that contain suggested token: Guided vs Base (non-guided).

Figure 6.8: The impact of a *"Favor"* suggestion token over each attribute in the feature category *"Browser"*.


Figure 6.9: Training dataset attribute occurrences of feature category *"Browser"* categorized by the positiveness of the strategy they occur in.

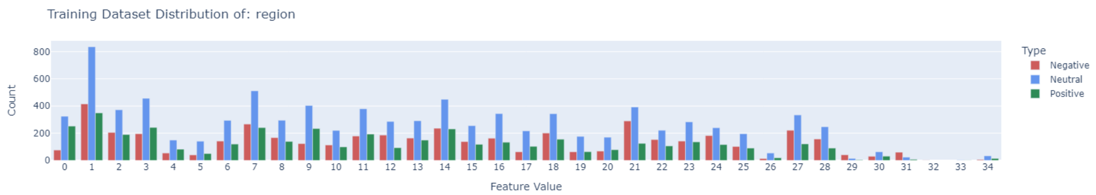
avoid certain attribute, which is why it is able to almost muting out all attributes except some that are present frequently in positive strategies in training data. Conversely, for some attributes that are rarely or never found in negative strategies, the impact of the *"Avoid"* token at inference is either negligible or paradoxically increases the output rate, which is opposite to the intended behavior.

#### 6.4.3.2 Multi-Token Analysis

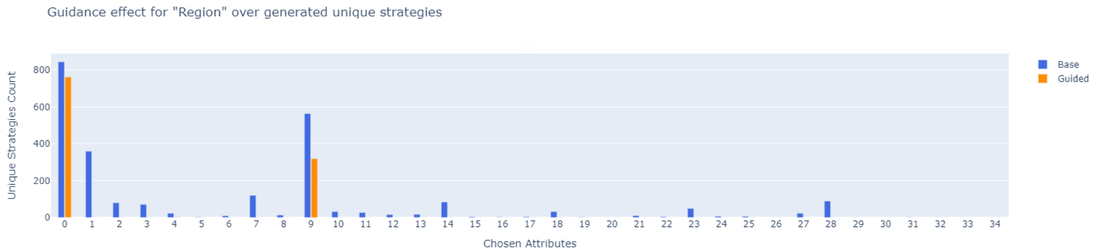
In scenarios involving suggestions of multiple attributes, we differentiate between two scenarios: multi-token single-feature and multi-token multi-feature. In the multi-token single-feature scenario, all suggestion tokens pertain to a single feature, functioning similarly to an OR condition. Conversely, the multi-token multi-feature scenario operates like an AND condition, with suggestions spanning across multiple features. Thus, our analysis follows the approach used in the single token scenario but modifies the counting methodology. Instead of merely counting occurrences, we also monitor whether all suggested attributes (AND) or any of the suggested attributes (OR) are present.

For the multi-token single-feature scenario, since only counting if any of the attributes is present is not sufficient to detect if a certain attribute is dominating the other suggested attributes. We propose to measure the membership ratio for each attribute.

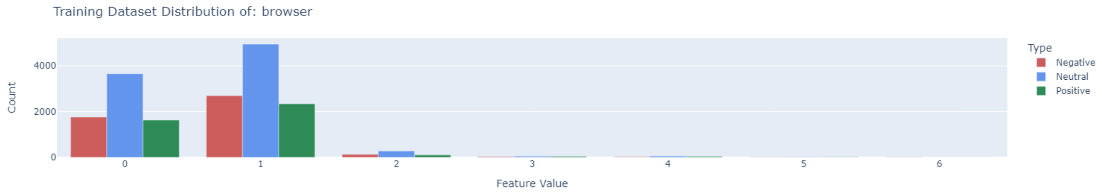
$$M(a) = \frac{\sum_{s^+ \in S^{favor} | p(c,s^+) \geqslant p_{min}} \mathbf{1}_{s^+}(a)}{\sum_{a' \in A'} \sum_{s^+ \in S^{favor} | p(c,s^+) \geqslant p_{min}} \mathbf{1}_{s^+}(a')} \tag{6.6}$$



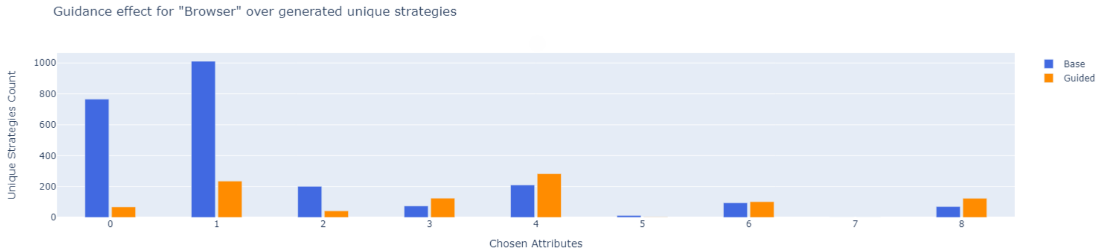

(a) Training dataset attribute occurrences of feature category *"Region"* categorized by the positiveness of the strategy they occur in.




(b) Comparison of unique strategies count that contain suggested token: Guided vs Base (non-guided).




(c) Training dataset attribute occurrences of feature category *"Browser"* categorized by the positiveness of the strategy they occur in.




(d) Comparison of unique strategies count that contain suggested token: Guided vs Base (non-guided).

Figure 6.10: The impact of a *"Avoid"* suggestion token over each attribute in the feature categories *"Region"* and *"Browser"*.

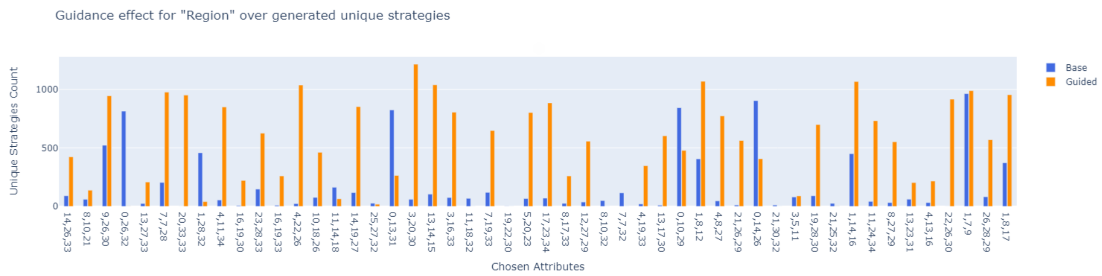



Figure 6.11: The impact of a *"Favor"* suggestion token in multi-attribute mode (OR) in the feature category *"Region"*.

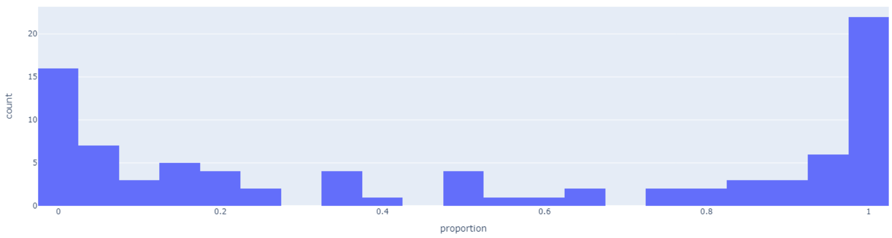



Figure 6.12: Distribution of membership ratios for 3 attributes of the *"Region"* feature category.

Such that $a$ is a suggested attribute and $A'$ is the set of suggested attributes. This measures the ratio of each suggested attribute in the set of strategies that adhere to suggested preferences.

Figure 6.11 shows *"Favor"* token guidance on a sample of *50* combinations of $x = 3$ attributes from the *"Region"* feature category. A strategy is counted as successfully guided if it contains one of the three suggested attributes. We note an effective guidance in most of the cases. However, to further analyze token guidance effectiveness in this multi-token scenario, we need to analyze membership rates $M(a)$ of all suggested attributes.

Figure 6.12 shows the distribution of the membership rates. A desired membership distribution should be centered around $1/x$, meaning that all suggested attributes appear evenly in the generated outputs. Unfortunately, in our sample we observe that the distribution of membership ratio is more or less U shaped, which means that some attributes are clearly dominant during generation even when multiple are favored.

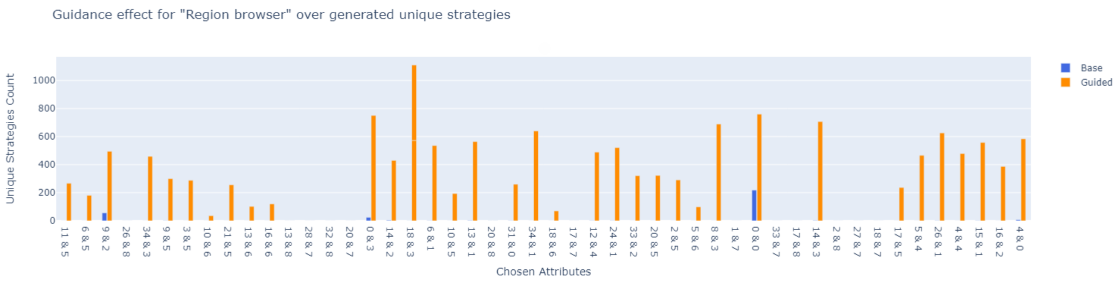



Figure 6.13: The impact of a *"Favor"* suggestion token in multi-feature mode (AND) in feature categories *"Region"* and *"Browser"*.

In the multi-token multi-feature scenario, which translates to an AND condition, we measure the uplift brought by the suggestion tokens when all of them are present. Similarly to the previous case, we use $x = 3$ *"Favor"* tokens over the *"Region"* and *"Browser"* feature categories. Figure 6.13 shows the count of unique strategies that contain all the suggested tokens distribution vs the generated unique strategies in standard mode (non-guided).

Table 6.2 provides detailed metrics for the generated strategies that accurately follow suggestions. It also includes results from two additional experiments involving multi-token multi-feature suggestions to assess the difficulty for the model in adhering to suggestions when suggestions become restrictive and options are limited. We observe that strategies aligning well with user preferences demonstrate strong positiveness and high data fidelity. Furthermore, there is a noticeable trend where increasing the number of suggested features significantly reduces the diversity and number of suggestion-compliant strategies, while also marginally decreasing data fidelity.

### 6.4.4 Discussion

This discussion highlights the balance between the novel capabilities introduced by our method and the challenges that need addressing. Going forward, it will be critical to address these limitations to enhance the model's capability to generate high performance

Table 6.2: Overall comparison between suggestion scenarios.

| Guidance | Features | Counts | Cosine | Hamming | Positiveness |
|---|---|---|---|---|---|
| Single-Token | Region | 931.28 | 0.886 | 0.183 | 0.781 |
| | Browser | 768.33 | 0.808 | 0.225 | 0.834 |
| Multi-Token Single-Feature | Region | 516.50 | 0.908 | 0.166 | 0.770 |
| Multi-Token Multi-Feature | Region & Browser | 289.48 | 0.817 | 0.261 | 0.783 |
| | Region & Browser & Os | 75.98 | 0.828 | 0.293 | 0.770 |
| | Region & Browser & Os & Gender | 10.50 | 0.770 | 0.323 | 0.766 |

personalized strategies while preserving a robust and effective control mechanism.

#### 6.4.4.1 Advantages

- **Targeted Favoring without Performance Compromise:** We proposed a novel approach that allows for the preference of certain outputs, termed as "favoring", without detracting from the overall performance of the model. This ensures that while specific attributes or strategies are prioritized, the quality of generation remains high.
- **Generalizability to Other Models:** The simplicity of the approach underpins its potential for generalization across various generative transformer-based models.
- **Extension to Multiple Special Embeddings:** The methodology is versatile enough to accommodate multiple special embeddings per feature category. This extension provides opportunities for a more nuanced control and customization of the generated outputs, improving the model's ability to adapt to intricate user preferences.

#### 6.4.4.2 Limitations

- **Vague Avoidance Guidance:** The current mechanism for strategy avoidance still suffers from a lack of specificity. The model's guidance on what to avoid is too imprecise, leading to *"Avoid"* embeddings that are poorly learned and not as effective as intended.

- **Discrete Signaling:** The signaling mechanism is discrete (*"Favor"*, *"Neutral"*, *"Avoid"*), offering no gradual level of suggestion. This discrete approach limits the model's ability to subtly adjust its output based on varying degrees of user preference intensity.
- **Dependence on Training Data:** The model's effectiveness in responding to *"Avoid"* tokens and generalizing avoidance across different attributes heavily relies on the training data distribution. If certain features are not represented in negative contexts within the training data, the model may struggle to apply avoidance effectively to these unseen scenarios. Augmenting the dataset artificially using the Estimator model could help mitigate this issue, but might hurt data fidelity.
- **Challenges in Evaluating Mixed Token Impact:** Assessing the impact of mixed token suggestions—where *"Favor"* and *"Avoid"* tokens are used simultaneously—presents a methodological challenge. Determining the net effect of conflicting signals on the generated output requires sophisticated measurement techniques to accurately capture the nuanced interplay between favoring and avoiding certain attributes.

## 6.5 Conclusion

In conclusion, this chapter has introduced a novel token-driven approach for integrating user preferences into the generation process of advertising strategies, offering a detailed examination of its application within the *ASGAR* model framework. Through a series of meticulously designed evaluations, we have demonstrated the method's efficiency in guiding the model towards generating desired outputs, while also identifying potential areas of improvement, particularly in handling negative preferences and refining how user preferences are integrated. We observed outstanding results, where the model demonstrated strong adherence to user suggestions without compromising performance, which is crucial in a production environment for meeting client needs.

The advantages of this approach, including its potential for generalization across different transformer-based generative models, highlight its value in advancing the field of personalized advertising strategy generation. However, the limitations encountered underscore the necessity for ongoing research and development.

# Chapter 7

# Conclusion and Future Directions

## Contents

This thesis presented several novel concepts and techniques to fill critical gaps in the domain of digital advertising strategy design through the integration of advanced deep learning methodologies. Over the course of six chapters, we navigated the intricate challenges and complexities that characterize the digital advertising landscape, ultimately presenting a comprehensive framework for AI-assisted generation and optimization of advertising campaigns.

In this chapter we provide a summary of the undertaken research issues (Section 7.1) and a summary of research outcomes (Section 7.2). We also discuss potential future research directions (Section 7.3).

## 7.1 Summary of the Research Issues

In conducting this research, we encountered a range of complex issues that reflect the nuanced and multifaceted nature of digital advertising strategy design. At the heart of these challenges was the need to navigate the vast and intricate feature space of advertising strategies, necessitating the use of advanced deep learning approaches to circumvent exhaustive and combinatorial exploration. Even filtering methods to bring the feature space to a more manageable size needed to be avoided, as we had no prior on the valuation of advertising strategies. This is due to the atomic nature of each strategy—where performance is intimately tied to the unique combination of features—which also makes the sequential generation of strategy elements inherently suboptimal, emphasizing the importance of order-agnostic processing. Therefore, it was critical to model feature interactions precisely. Accurately modeling and predicting the performance of diverse advertising strategies also highlighted the essential need for data fidelity, ensuring outputs are trustworthy and closely aligned with known data points.

In laying the groundwork for the development of an AI-assistant for advertising strategy design, alongside contextual strict conditioning, we focused on accommodating user preferences as flexible suggestions rather than rigid directives. This approach allows for the inclusion of user-suggested preferences as desirable outputs when they lead to effective strategies and disregarding them when they do not, in order to produce superior strategies. This brought a new set of challenges and required the developement of a novel approach to address them.

A recurring issue we faced was mode collapse, where the model converges on generating a limited subset of strategies or sometimes even a single strategy, resulting in insufficiently diverse outputs. This issue was challenging to address, requiring precise control over the variational aspects of the model and tailored loss functions. These adaptations were necessary to balance information compression and prevent input leakage, ensuring a broader range of output strategies.

Addressing these challenges demanded a blend of technical innovation, meticulous data analysis, and informed application of domain-specific knowledge, highlighting the intricate factors that influence both the training of the model and the effectiveness of the generated advertising campaigns. Evaluating the effectiveness of our methods has also been challenging, as it involves balancing and interpreting multiple performance metrics, each providing only a partial view of the overall performance. This necessitates the development of rigorous evaluation protocols tailored to this recommendation and generative task.

## 7.2 Summary of the Research Outcomes

The first contribution, detailed in Chapter 4, presents what we believe to be the first approach that specifically addresses the critical constraints inherent in digital advertising strategy design. We tackled the challenge of combinatorial explosion by introducing a transformer-based model within a new framework for guided non-autoregressive generation. We also introduced a novel loss function that facilitates guided exploration of the feature space through smooth contrastive learning, complete with hyperparameters that allow customization of the generated strategies' similarity to the dataset's successful strategies. A VAE-like variational component and consistent dropout use were employed to enhance output diversity. Given the lack of directly comparable methods, we benchmarked our results against prominent methods from other fields, adapted for this specific task. This contribution demonstrated superior results, outperforming other approaches while adhering to the task's constraints and providing adjustable exploration/exploitation options essential for meeting diverse client needs.

The second contribution, detailed in Chapter 5, aimed to enhance output diversity and robustness in our methodology. We improved our initial approach via an exploratory mode activated at inference and proposed a new variational component with vector quantization techniques, which increased generative diversity and mitigated mode collapse. To address a key limitation identified in our first model—the strategy dissimilarity function—we implemented a learned metric network, which when combined with cross-entropy improved the model's ability to distinguish between seemingly similar strategies that vary significantly in performance. We also refined our evaluation protocol by introducing new metrics and processes to more accurately assess the effectiveness and quality of our methods. As a result, we effectively mitigated mode collapse and achieved significantly enhanced generative diversity. The exploratory mode at inference time performed well, delivering excellent results without compromising overall performance, proving crucial for clients seeking to explore new strategies without retraining the entire framework. The improved evaluation protocol confirmed through clearer performance metrics the effectiveness of our methods and provided deeper insights to refine the training process.

The third contribution, detailed in Chapter 6, focused on developing an AI assistant for advertising strategy design with the aim of integrating user preferences as flexible guidelines rather than strict directives. Previously, certain feature attributes were treated as fixed context, imposing rigid conditions; now, other attributes might be considered as desirable outcomes but only if they contribute to high-performance strategies, effectively acting as suggestions. Building on our earlier work, we developed a novel token-driven approach that incorporates user preferences into a transformer-based generative model using specialized token embeddings. We introduced a scheduled token masking strategy during training, which helped the model learn these suggestive token embeddings. In practice, these tokens guide the model to either consider user suggestions or to operate based on its existing knowledge. We also modified the loss function to prevent newly identified instances of mode collapse, particularly when certain attributes are suggested against (avoided). Extensive experiments were conducted to evaluate this approach, which produced outstanding results, demonstrating its viability across various domains utilizing transformer models. The model demonstrated strong adherence to user suggestions without compromising performance, which is crucial in a production environment for meeting client needs.

Our research makes significant contributions to both academia and the digital advertising industry by introducing innovative generative methods. Academically, it advances AI applications in marketing research by offering new methods for customizing and optimizing digital campaigns and challenging existing paradigms in model training. To the best of our knowledge, this study is the first to address the primary constraints of advertising strategy design comprehensively, proposing a series of methodological enhancements. Industrially, it has the potential to transform digital advertising by enabling more optimized and efficient campaigns through AI, reducing the need for intensive resource allocation and facilitating dynamic, AI-assisted strategy design. Additionally, our work has the potential to pioneer AI-assisted media planning through generative models, a concept that has not yet been proposed in the industry at the time of writing this manuscript.

## 7.3 Future Research Directions

Our contributions, while advancing the field, highlight areas ready for further exploration and improvement. Throughout the chapters, we pinpointed several limitations that, if addressed, could enhance performance significantly.

These improvements include developing a better strategy similarity function to further stabilize training and reduce instances of confusing signals where the model perceives strategies as similar while they vastly different in performance. There is also a need for a mechanism that allows for a finer balance between exploration and exploitation during inference. Additionally, a more robust solution is needed to address the recurring issue of mode collapse, particularly in exploration mode where the model, lacking sufficient anchor examples, tends to default to a single answer. Furthermore, improving the suggestive token mechanism to allow for weighted constraints would enable the expression of more subtle and nuanced preferences.

Working towards explainability and trustworthiness, the evaluation protocols also need refinement to enhance understanding of approach effectiveness and identify potential issues. Additionally, optimizing resource usage is crucial as transformer models become more resource-intensive. Improving the model architecture could reduce reliance on extensive datasets and help mitigate data bias, common issues in digital advertising datasets.

These points emphasize the need for continuous refinement of our methods and open up wide-ranging opportunities for future research in the digital marketing landscape.

Furthermore, our methods are generalizable to other fields, particularly our suggestive token mechanism, which could enhance personalization in AI models such as large language models (LLMs) and quantization-based image generative models, especially those based on transformers. This adaptability could be particularly valuable for addressing ethical and fairness considerations in AI applications.

## 7.4 Personal Reflections

This thesis was nice[1].

[1]Full of life, enriching, challenging, rewarding, stressful, giving purpose... It has offered me a unique opportunity to delve into the complexities of machine learning, pushing the boundaries of my understanding and capabilities. The process of navigating through the vast literature, experimenting with intricate models, and contributing to the field has fostered a deep appreciation for the power and potential of deep learning. It has been a period of significant personal and professional growth, filled with moments of reflection, failure, stubbornness, breakthrough and realization. Reflecting on this journey, I feel a sense of accomplishment and a renewed curiosity and deepened interest, eager to continue exploring the beautiful clash of math and computer science that deep learning presents.

# Bibliography

[Abdullah et al., 2022] Abdullah, M., Madain, A., and Jararweh, Y. (2022). ChatGPT: Fundamentals, Applications and Social Impacts. In Ceravolo, P., Guetl, C., Jararweh, Y., and Benkhelifa, E., editors, *Ninth International Conference on Social Networks Analysis, Management and Security, SNAMS 2022, Milan, Italy, November 29 - Dec. 1, 2022*, pages 1–8. IEEE.

[Amazon Ads, 2024] Amazon Ads (2024). DSP vs SSP vs DMP—What is the difference? `https://advertising.amazon.com/blog/dsp-ssp-dmp-difference?ref_=a20m_us_fnav_lngsw_en-us`. Accessed: 2024-06-25.

[Arroyo et al., 2021] Arroyo, D. M., Postels, J., and Tombari, F. (2021). Variational Transformer Networks for Layout Generation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 13642–13652. Computer Vision Foundation / IEEE.

[Avadhanula et al., 2021] Avadhanula, V., Colini-Baldeschi, R., Leonardi, S., Sankararaman, K. A., and Schrijvers, O. (2021). Stochastic bandits for multi-platform budget optimization in online advertising. In Leskovec, J., Grobelnik, M., Najork, M., Tang, J., and Zia, L., editors, *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 2805–2817. ACM / IW3C2.

[Bai et al., 2019] Bai, J., Zhou, C., Song, J., Qu, X., An, W., Li, Z., and Gao, J. (2019). Personalized Bundle List Recommendation. In Liu, L., White, R. W., Mantrach, A., Silvestri, F., McAuley, J. J., Baeza-Yates, R., and Zia, L., editors, *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 60–71. ACM.

[Ban et al., 2021] Ban, Y., He, J., and Cook, C. B. (2021). Multi-facet Contextual Bandits: A Neural Network Perspective. In Zhu, F., Ooi, B. C., and Miao, C., editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 35–45. ACM.

[Bao et al., 2023] Bao, K., Zhang, J., Zhang, Y., Wang, W., Feng, F., and He, X. (2023). TALLRec: An Effective and Efficient Tuning Framework to Align Large Language Model with Recommendation. In Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T. D., Basilico, J., Pizzato, L., and Song, Y., editors, *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 1007–1014. ACM.

[Beheshtian-Ardakani et al., 2018] Beheshtian-Ardakani, A., Fathian, M., and Gholamian, M. R. (2018). A novel model for product bundling and direct marketing in e-commerce based on market segmentation. *Decision Science Letters*, 7:39–54.

[Beltagy et al., 2020] Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The Long-Document Transformer. *CoRR*, abs/2004.05150.

[Benamara and Viennet, 2023] Benamara, I. and Viennet, E. (2023). Contextual Advertising Strategy Generation via Attention and Interaction Guidance. In *10th IEEE International Conference on Data Science and Advanced Analytics, DSAA 2023, Thessaloniki, Greece, October 9-13, 2023*, pages 1–10. IEEE.

[Bengio et al., 2013] Bengio, Y., Léonard, N., and Courville, A. C. (2013). Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation. *CoRR*, abs/1308.3432.

[Betker et al., 2023] Betker, J., Goh, G., Jing, L., TimBrooks, Wang, J., Li, L., LongOuyang, JuntangZhuang, JoyceLee, YufeiGuo, WesamManassra, PrafullaDhariwal, CaseyChu, YunxinJiao, and Ramesh, A. (2023). Improving Image Generation with Better Captions. *OpenAI*.

[Beutel et al., 2018] Beutel, A., Covington, P., Jain, S., Xu, C., Li, J., Gatto, V., and Chi, E. H. (2018). Latent Cross: Making Use of Context in Recurrent Recommender Systems. In Chang, Y., Zhai, C., Liu, Y., and Maarek, Y., editors, *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, pages 46–54. ACM.

[Bian et al., 2022] Bian, W., Wu, K., Ren, L., Pi, Q., Zhang, Y., Xiao, C., Sheng, X., Zhu, Y., Chan, Z., Mou, N., Luo, X., Xiang, S., Zhou, G., Zhu, X., and Deng, H. (2022). CAN: Feature Co-Action Network for Click-Through Rate Prediction. In Candan, K. S., Liu, H., Akoglu, L., Dong, X. L., and Tang, J., editors, *WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, pages 57–65. ACM.

[Bibas et al., 2023] Bibas, K., Shalom, O. S., and Jannach, D. (2023). Semi-supervised Adversarial Learning for Complementary Item Recommendation. In Ding, Y., Tang, J., Sequeda, J. F., Aroyo, L., Castillo, C., and Houben, G., editors, *Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023*, pages 1804–1812. ACM.

[Blakeman, 2015] Blakeman, R. (2015). *Advertising campaign design: Just the essentials*. Routledge.

[Blondel et al., 2016] Blondel, M., Fujino, A., Ueda, N., and Ishihata, M. (2016). Higher-Order Factorization Machines. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

[Brauwers and Frasincar, 2023] Brauwers, G. and Frasincar, F. (2023). A General Survey on Attention Mechanisms in Deep Learning. *IEEE Trans. Knowl. Data Eng.*, 35(4):3279–3298.

[Brosh et al., 2022] Brosh, T. A., Livne, A., Shalom, O. S., Shapira, B., and Last, M. (2022). BRUCE: Bundle Recommendation Using Contextualized item Embeddings. In Golbeck, J., Harper, F. M., Murdock, V., Ekstrand, M. D., Shapira, B., Basilico, J., Lundgaard, K. T., and Oldridge, E., editors, *RecSys '22: Sixteenth ACM Conference on Recommender Systems, Seattle, WA, USA, September 18 - 23, 2022*, pages 237–245. ACM.

[Brown et al., 2020] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language Models are Few-Shot Learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.*

[Bulatov et al., 2023] Bulatov, A., Kuratov, Y., and Burtsev, M. S. (2023). Scaling Transformer to 1M tokens and beyond with RMT. *CoRR*, abs/2304.11062.

[Cai et al., 2017] Cai, H., Ren, K., Zhang, W., Malialis, K., Wang, J., Yu, Y., and Guo, D. (2017). Real-Time Bidding by Reinforcement Learning in Display Advertising. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, page 661–670, New York, NY, USA. Association for Computing Machinery.

[Cao et al., 2017] Cao, D., Nie, L., He, X., Wei, X., Zhu, S., and Chua, T. (2017). Embedding Factorization Models for Jointly Recommending Items and User Generated Lists. In Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A. P., and White, R. W., editors, *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 585–594. ACM.

[Cao et al., 2023] Cao, M., Wang, X., Qi, Z., Shan, Y., Qie, X., and Zheng, Y. (2023). MasaCtrl: Tuning-Free Mutual Self-Attention Control for Consistent Image Synthesis and Editing. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 22503–22513. IEEE.

[Chang et al., 2022] Chang, H., Zhang, H., Jiang, L., Liu, C., and Freeman, W. T. (2022). MaskGIT: Masked Generative Image Transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 11305–11315. IEEE.

[Chang et al., 2020] Chang, J., Gao, C., He, X., Jin, D., and Li, Y. (2020). Bundle Recommendation with Graph Convolutional Networks. In Huang, J. X., Chang, Y., Cheng, X., Kamps, J., Murdock, V., Wen, J., and Liu, Y., editors, *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, pages 1673–1676. ACM.

[Chang et al., 2023] Chang, J., Gao, C., He, X., Jin, D., and Li, Y. (2023). Bundle Recommendation and Generation With Graph Neural Networks. *IEEE Trans. Knowl. Data Eng.*, 35(3):2326–2340.

[Chen et al., 2016] Chen, J., Sun, B., Li, H., Lu, H., and Hua, X. (2016). Deep CTR Prediction in Display Advertising. In Hanjalic, A., Snoek, C., Worring, M., Bulterman, D. C. A., Huet, B., Kelliher, A., Kompatsiaris, Y., and Li, J., editors, *Proceedings of the 2016 ACM Conference on Multimedia Conference, MM 2016, Amsterdam, The Netherlands, October 15-19, 2016*, pages 811–820. ACM.

[Chen et al., 2017] Chen, J., Zhang, H., He, X., Nie, L., Liu, W., and Chua, T. (2017). Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention. In Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A. P., and White, R. W., editors, *Proceedings of the 40th International ACM SIGIR*

*Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 335–344. ACM.

[Chen et al., 2019a] Chen, L., Liu, Y., He, X., Gao, L., and Zheng, Z. (2019a). Matching User with Item Set: Collaborative Bundle Recommendation with Deep Attention Network. In Kraus, S., editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 2095–2101. ijcai.org.

[Chen et al., 2019b] Chen, W., Huang, P., Xu, J., Guo, X., Guo, C., Sun, F., Li, C., Pfadler, A., Zhao, H., and Zhao, B. (2019b). POG: Personalized Outfit Generation for Fashion Recommendation at Alibaba iFashion. In Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., and Karypis, G., editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2662–2670. ACM.

[Chen et al., 2011] Chen, Y., Berkhin, P., Anderson, B., and Devanur, N. R. (2011). Real-time bidding algorithms for performance-based display ad allocation. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, page 1307–1315, New York, NY, USA. Association for Computing Machinery.

[Chen, 2023] Chen, Z. (2023). PALR: Personalization Aware LLMs for Recommendation. *CoRR*, abs/2305.07622.

[Cheng et al., 2016] Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., Anil, R., Haque, Z., Hong, L., Jain, V., Liu, X., and Shah, H. (2016). Wide & Deep Learning for Recommender Systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, DLRS 2016, page 7–10, New York, NY, USA. Association for Computing Machinery.

[Cheng et al., 2019] Cheng, P., Liu, C., Li, C., Shen, D., Henao, R., and Carin, L. (2019). Straight-Through Estimator as Projected Wasserstein Gradient Flow. *CoRR*, abs/1910.02176.

[Cheng et al., 2020] Cheng, W., Shen, Y., and Huang, L. (2020). Adaptive Factorization Network: Learning Adaptive-Order Feature Interactions. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 3609–3616. AAAI Press.

[Choi et al., 2018] Choi, Y., Choi, M., Kim, M., Ha, J., Kim, S., and Choo, J. (2018). StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8789–8797. Computer Vision Foundation / IEEE Computer Society.

[Chopra et al., 2005] Chopra, S., Hadsell, R., and LeCun, Y. (2005). Learning a Similarity Metric Discriminatively, with Application to Face Verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, pages 539–546. IEEE Computer Society.

[Cipolla et al., 2018] Cipolla, R., Gal, Y., and Kendall, A. (2018). Multi-task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7482–7491, Los Alamitos, CA, USA. IEEE Computer Society.

[Copet et al., 2023] Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., and Défossez, A. (2023). Simple and Controllable Music Generation. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems 36: Annual Conference on*

*Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023.*

[Covington et al., 2016a] Covington, P., Adams, J., and Sargin, E. (2016a). Deep Neural Networks for YouTube Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, page 191–198, New York, NY, USA. Association for Computing Machinery.

[Covington et al., 2016b] Covington, P., Adams, J., and Sargin, E. (2016b). Deep Neural Networks for YouTube Recommendations. In Sen, S., Geyer, W., Freyne, J., and Castells, P., editors, *Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, September 15-19, 2016*, pages 191–198. ACM.

[Dai et al., 2023] Dai, S., Shao, N., Zhao, H., Yu, W., Si, Z., Xu, C., Sun, Z., Zhang, X., and Xu, J. (2023). Uncovering ChatGPT's Capabilities in Recommender Systems. In Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T. D., Basilico, J., Pizzato, L., and Song, Y., editors, *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 1126–1132. ACM.

[Deng et al., 2021] Deng, Q., Wang, K., Zhao, M., Wu, R., Ding, Y., Zou, Z., Shang, Y., Tao, J., and Fan, C. (2021). Build Your Own Bundle - A Neural Combinatorial Optimization Method. In Shen, H. T., Zhuang, Y., Smith, J. R., Yang, Y., César, P., Metze, F., and Prabhakaran, B., editors, *MM '21: ACM Multimedia Conference, Virtual Event, China, October 20 - 24, 2021*, pages 2625–2633. ACM.

[Deng et al., 2020] Deng, Q., Wang, K., Zhao, M., Zou, Z., Wu, R., Tao, J., Fan, C., and Chen, L. (2020). Personalized Bundle Recommendation in Online Games. In d'Aquin, M., Dietze, S., Hauff, C., Curry, E., and Cudré-Mauroux, P., editors, *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 2381–2388. ACM.

[Devlin et al., 2019] Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

[Dhariwal and Nichol, 2021] Dhariwal, P. and Nichol, A. Q. (2021). Diffusion Models Beat GANs on Image Synthesis. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 8780–8794.

[Dilbaz and Saribas, 2023] Dilbaz, S. and Saribas, H. (2023). STEC: See-Through Transformer-based Encoder for CTR Prediction. *CoRR*, abs/2308.15033.

[Dong et al., 2022] Dong, Z., Wei, P., and Lin, L. (2022). DreamArtist: Towards Controllable One-Shot Text-to-Image Generation via Contrastive Prompt-Tuning. *CoRR*, abs/2211.11337.

[Dosovitskiy et al., 2021] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

[Du et al., 2017] Du, M., Sassioui, R., Varisteas, G., State, R., Brorsson, M., and Cherkaoui, O. (2017). Improving Real-Time Bidding Using a Constrained Markov Decision Process. In *Advanced Data Mining and Applications*, pages 711–726, Cham. Springer International Publishing.

[El-Nouby et al., 2023] El-Nouby, A., Muckley, M. J., Ullrich, K., Laptev, I., Verbeek, J., and Jégou, H. (2023). Image Compression with Product Quantized Masked Image Modeling. *Trans. Mach. Learn. Res.*, 2023.

[Esser et al., 2021] Esser, P., Rombach, R., and Ommer, B. (2021). Taming Transformers for High-Resolution Image Synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 12873–12883. Computer Vision Foundation / IEEE.

[Fan et al., 2018] Fan, A., Lewis, M., and Dauphin, Y. (2018). Hierarchical Neural Story Generation. In Gurevych, I. and Miyao, Y., editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

[Fan and Delage, 2022] Fan, R. and Delage, E. (2022). Risk-Aware Bid Optimization for Online Display Advertisement. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, CIKM '22. ACM.

[Fang et al., 2021] Fang, L., Zeng, T., Liu, C., Bo, L., Dong, W., and Chen, C. (2021). Transformer-based Conditional Variational Autoencoder for Controllable Story Generation. *CoRR*, abs/2101.00828.

[Fang et al., 2018] Fang, Y., Xiao, X., Wang, X., and Lan, H. (2018). Customized Bundle Recommendation by Association Rules of Product Categories for Online Supermarkets. In *Third IEEE International Conference on Data Science in Cyberspace, DSC 2018, Guangzhou, China, June 18-21, 2018*, pages 472–475. IEEE.

[Frolov and Oseledets, 2017] Frolov, E. and Oseledets, I. V. (2017). Tensor methods and recommender systems. *WIREs Data Mining Knowl. Discov.*, 7(3).

[Ghosh et al., 2019] Ghosh, A., Mitra, S., Sarkhel, S., Xie, J., Wu, G., and Swaminathan, V. (2019). Scalable Bid Landscape Forecasting in Real-Time Bidding. In Brefeld, U., Fromont, É., Hotho, A., Knobbe, A. J., Maathuis, M. H., and Robardet, C., editors, *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Würzburg, Germany, September 16-20, 2019, Proceedings, Part III*, volume 11908 of *Lecture Notes in Computer Science*, pages 451–466. Springer.

[Gong et al., 2019] Gong, Y., Zhu, Y., Duan, L., Liu, Q., Guan, Z., Sun, F., Ou, W., and Zhu, K. Q. (2019). Exact-K Recommendation via Maximal Clique Optimization. In Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., and Karypis, G., editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 617–626. ACM.

[Goodfellow et al., 2014] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. (2014). Generative Adversarial Networks. *CoRR*, abs/1406.2661.

[Google, 2024a] Google (2024a). About line items - Google Ad Manager Help. `https://support.google.com/admanager/answer/9405477?hl=en&sjid=10634286061942915087-EU&visit_id=638548756046725834-3678005709&ref_topic=7506394&rd=1`. Accessed: 2024-06-25.

[Google, 2024b] Google (2024b). FAQs about AdSense moving to a first-price auction - Google AdSense Help. `https://support.google.com/adsense/answer/10858748?hl=en`. [Accessed 25-06-2024].

[Grislain et al., 2019] Grislain, N., Perrin, N., and Thabault, A. (2019). Recurrent Neural Networks for Stochastic Control in Real-Time Bidding. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19. ACM.

[Gu and Dao, 2023] Gu, A. and Dao, T. (2023). Mamba: Linear-Time Sequence Modeling with Selective State Spaces. *CoRR*, abs/2312.00752.

[Gu et al., 2022] Gu, A., Goel, K., and Ré, C. (2022). Efficiently Modeling Long Sequences with Structured State Spaces. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

[Guan et al., 2021] Guan, Z., Wu, H., Cao, Q., Liu, H., Zhao, W., Li, S., Xu, C., Qiu, G., Xu, J., and Zheng, B. (2021). Multi-Agent Cooperative Bidding Games for Multi-Objective Optimization in e-Commercial Sponsored Search. In Zhu, F., Ooi, B. C., and Miao, C., editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 2899–2909. ACM.

[Guo et al., 2017] Guo, H., Tang, R., Ye, Y., Li, Z., and He, X. (2017). DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In Sierra, C., editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 1725–1731. ijcai.org.

[Guo et al., 2021] Guo, L., Jin, J., Zhang, H., Zheng, Z., Yang, Z., Xing, Z., Pan, F., Niu, L., Wu, F., Xu, H., Yu, C., Jiang, Y., and Zhu, X. (2021). We Know What You Want: An Advertising Strategy Recommender System for Online Advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 2919–2927, New York, NY, USA. Association for Computing Machinery.

[Harte et al., 2023] Harte, J., Zorgdrager, W., Louridas, P., Katsifodimos, A., Jannach, D., and Fragkoulis, M. (2023). Leveraging Large Language Models for Sequential Recommendation. In Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T. D., Basilico, J., Pizzato, L., and Song, Y., editors, *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 1096–1102. ACM.

[He and Chua, 2017] He, X. and Chua, T. (2017). Neural Factorization Machines for Sparse Predictive Analytics. In Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A. P., and White, R. W., editors, *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 355–364. ACM.

[He et al., 2017] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T. (2017). Neural Collaborative Filtering. In Barrett, R., Cummings, R., Agichtein, E., and Gabrilovich, E., editors, *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 173–182. ACM.

[He et al., 2019] He, Y., Wang, J., Niu, W., and Caverlee, J. (2019). A Hierarchical Self-Attentive Model for Recommending User-Generated Item Lists. In Zhu, W., Tao, D., Cheng, X., Cui, P., Rundensteiner, E. A., Carmel, D., He, Q., and Yu, J. X., editors, *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 1481–1490. ACM.

[He et al., 2022] He, Z., Zhao, H., Yu, T., Kim, S., Du, F., and McAuley, J. J. (2022). Bundle MCR: Towards Conversational Bundle Recommendation. In Golbeck, J., Harper, F. M., Murdock, V., Ekstrand, M. D., Shapira, B., Basilico, J., Lundgaard, K. T., and Oldridge, E., editors, *RecSys '22: Sixteenth ACM Conference on Recommender Systems, Seattle, WA, USA, September 18 - 23, 2022*, pages 288–298. ACM.

[Hertz et al., 2023] Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., and Cohen-Or, D. (2023). Prompt-to-Prompt Image Editing with Cross-Attention Control. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

[Holtzman et al., 2020] Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. (2020). The Curious Case of Neural Text Degeneration. In *8th International Conference* *on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

[Hong and Jung, 2018] Hong, M. and Jung, J. J. (2018). Multi-Sided recommendation based on social tensor factorization. *Inf. Sci.*, 447:140–156.

[Hu and He, 2019] Hu, H. and He, X. (2019). Sets2Sets: Learning from Sequential Sets with Neural Networks. In Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., and Karypis, G., editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 1491–1499. ACM.

[Hu et al., 2022] Hu, J., Yi, X., Li, W., Sun, M., and Xie, X. (2022). Fuse It More Deeply! A Variational Transformer with Layer-Wise Latent Variable Inference for Text Generation. In Carpuat, M., de Marneffe, M., and Ruíz, I. V. M., editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 697–716. Association for Computational Linguistics.

[Huang et al., 2019] Huang, T., Zhang, Z., and Zhang, J. (2019). FiBiNET: combining feature importance and bilinear feature interaction for click-through rate prediction. In *Proceedings of the 13th ACM Conference on Recommender Systems*, RecSys '19. ACM.

[Hughes et al., 2019] Hughes, J. W., Chang, K., and Zhang, R. (2019). Generating Better Search Engine Text Advertisements with Deep Reinforcement Learning. In Teredesai, A., Kumar, V., Li, Y., Rosales, R., Terzi, E., and Karypis, G., editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2269–2277. ACM.

[IAB, 2023] IAB (2023). Internet Advertising Revenue Report: Full Year 2022.

[Ilias and Askounis, 2023] Ilias, L. and Askounis, D. (2023). Context-aware attention layers coupled with optimal transport domain adaptation and multimodal fusion methods for recognizing dementia from spontaneous speech. *Knowl. Based Syst.*, 277:110834.

[Jang et al., 2017] Jang, E., Gu, S., and Poole, B. (2017). Categorical Reparameterization with Gumbel-Softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

[Jiang et al., 2023] Jiang, Z., Zhou, K., Zhang, M., Chen, R., Hu, X., and Choi, S. (2023). Adaptive RiskAware Bidding with Budget Constraint in Display Advertising. *SIGKDD Explor. Newsl.*, 25(1):73–82.

[Jin et al., 2018] Jin, J., Song, C., Li, H., Gai, K., Wang, J., and Zhang, W. (2018). Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18. ACM.

[Juan et al., 2016] Juan, Y., Zhuang, Y., Chin, W.-S., and Lin, C.-J. (2016). Field-aware Factorization Machines for CTR Prediction. In *Proceedings of the 10th ACM Conference on Recommender Systems*, RecSys '16, page 43–50, New York, NY, USA. Association for Computing Machinery.

[Kamigaito et al., 2021] Kamigaito, H., Zhang, P., Takamura, H., and Okumura, M. (2021). An Empirical Study of Generating Texts for Search Engine Advertising. In Kim, Y., Li, Y., and Rambow, O., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 255–262. Association for Computational Linguistics.

[Karras et al., 2019] Karras, T., Laine, S., and Aila, T. (2019). A Style-Based Generator Architecture for Generative Adversarial Networks. In *IEEE Conference on Computer*

*Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4401–4410. Computer Vision Foundation / IEEE.

[Katz et al., 2022] Katz, O., Barkan, O., Koenigstein, N., and Zabari, N. (2022). Learning to Ride a Buy-Cycle: A Hyper-Convolutional Model for Next Basket Repurchase Recommendation. In Golbeck, J., Harper, F. M., Murdock, V., Ekstrand, M. D., Shapira, B., Basilico, J., Lundgaard, K. T., and Oldridge, E., editors, *RecSys '22: Sixteenth ACM Conference on Recommender Systems, Seattle, WA, USA, September 18 - 23, 2022*, pages 316–326. ACM.

[Kaya and Bilge, 2019] Kaya, M. and Bilge, H. S. (2019). Deep Metric Learning: A Survey. *Symmetry*, 11(9):1066.

[Keskar et al., 2019] Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., and Socher, R. (2019). CTRL: A Conditional Transformer Language Model for Controllable Generation. *CoRR*, abs/1909.05858.

[Khosla et al., 2020] Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020). Supervised Contrastive Learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.*

[Kingma and Ba, 2015] Kingma, D. P. and Ba, J. (2015). Adam: A Method for Stochastic Optimization. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.*

[Kingma and Dhariwal, 2018] Kingma, D. P. and Dhariwal, P. (2018). Glow: Generative Flow with Invertible 1x1 Convolutions. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Pro-*

*cessing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 10236–10245.

[Kingma and Welling, 2014] Kingma, D. P. and Welling, M. (2014). Auto-Encoding Variational Bayes. In Bengio, Y. and LeCun, Y., editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

[Kolesnikov et al., 2022] Kolesnikov, A., Pinto, A. S., Beyer, L., Zhai, X., Harmsen, J., and Houlsby, N. (2022). UViM: A Unified Modeling Approach for Vision with Learned Guiding Codes. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

[Köpf et al., 2023] Köpf, A., Kilcher, Y., von Rütte, D., Anagnostidis, S., Tam, Z. R., Stevens, K., Barhoum, A., Nguyen, D., Stanley, O., Nagyfi, R., ES, S., Suri, S., Glushkov, D., Dantuluri, A., Maguire, A., Schuhmann, C., Nguyen, H., and Mattick, A. (2023). OpenAssistant Conversations - Democratizing Large Language Model Alignment. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

[Ku et al., 2023] Ku, Y., Kuznetsov, M., Mishra, S., and de Juan, P. (2023). Staging E-Commerce Products for Online Advertising using Retrieval Assisted Image Generation. In Bagherjeiran, A., Djuric, N., Lee, K., Pang, L., Radosavljevic, V., and Rajan, S., editors, *Proceedings of the Workshop on Data Mining for Online Advertising (AdKDD 2023) co-located with the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD 2023), Long Beach, CA, USA, August 7, 2023*, volume 3556 of *CEUR Workshop Proceedings*. CEUR-WS.org.

[Kumari et al., 2023] Kumari, N., Zhang, B., Zhang, R., Shechtman, E., and Zhu, J. (2023). Multi-Concept Customization of Text-to-Image Diffusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 1931–1941. IEEE.

[Lancucki et al., 2020] Lancucki, A., Chorowski, J., Sanchez, G., Marxer, R., Chen, N., Dolfing, H. J. G. A., Khurana, S., Alumäe, T., and Laurent, A. (2020). Robust Training of Vector Quantized Bottleneck Models. In *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*, pages 1–7. IEEE.

[Lee et al., 2022] Lee, D., Kim, C., Kim, S., Cho, M., and Han, W. (2022). Autoregressive Image Generation using Residual Quantization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 11513–11522. IEEE.

[Lee et al., 2013] Lee, K., Jalali, A., and Dasdan, A. (2013). Real time bid optimization with smooth budget delivery in online advertising. In Saka, E., Shen, D., Gao, B., Yan, J., and Li, Y., editors, *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising, ADKDD '13, Chicago, Illinois, USA, August 11, 2013*, pages 1:1–1:9. ACM.

[Li et al., 2010] Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In Rappa, M., Jones, P., Freire, J., and Chakrabarti, S., editors, *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, pages 661–670. ACM.

[Li et al., 2023] Li, X., Xi, W., Xing, X., and Wang, C. (2023). Auto Graph Filtering for Bundle Recommendation. In Chen, G., Khan, L., Gao, X., Qiu, M., Pedrycz, W., and Wu, X., editors, *IEEE International Conference on Data Mining, ICDM 2023, Shanghai, China, December 1-4, 2023*, pages 299–308. IEEE.

[Li et al., 2022] Li, X., Zhang, M. M., Wang, Z., and Tong, Y. (2022). Arbitrary Distribution Modeling with Censorship in Real-Time Bidding Advertising. In Zhang, A. and Rangwala, H., editors, *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, pages 3250–3258. ACM.

[Lian et al., 2018] Lian, J., Zhou, X., Zhang, F., Chen, Z., Xie, X., and Sun, G. (2018). xDeepFM: Combining Explicit and Implicit Feature Interactions for Recommender Systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18. ACM.

[Liao et al., 2014] Liao, H., Peng, L., Liu, Z., and Shen, X. (2014). iPinYou Global RTB Bidding Algorithm Competition Dataset. In *Proceedings of the Eighth International Workshop on Data Mining for Online Advertising*, ADKDD'14, page 1–6, New York, NY, USA. Association for Computing Machinery.

[Lin et al., 2020a] Lin, C.-C., Chuang, K.-T., Wu, W. C.-H., and Chen, M.-S. (2020a). Budget-Constrained Real-Time Bidding Optimization: Multiple Predictors Make It Better. *ACM Trans. Knowl. Discov. Data*, 14(2).

[Lin et al., 2023] Lin, J., Zhou, M., Ma, Y., Gao, Y., Fei, C., Chen, Y., Yu, Z., and Ge, T. (2023). AutoPoster: A Highly Automatic and Content-aware Design System for Advertising Poster Generation. In El-Saddik, A., Mei, T., Cucchiara, R., Bertini, M., Vallejo, D. P. T., Atrey, P. K., and Hossain, M. S., editors, *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*, pages 1250–1260. ACM.

[Lin et al., 2020b] Lin, Z., Winata, G. I., Xu, P., Liu, Z., and Fung, P. (2020b). Variational Transformers for Diverse Response Generation. *CoRR*, abs/2003.12738.

[Liu et al., 2020a] Liu, B., Xue, N., Guo, H., Tang, R., Zafeiriou, S., He, X., and Li, Z. (2020a). AutoGroup: Automatic Feature Grouping for Modelling Explicit High-Order Feature Interactions in CTR Prediction. In *Proceedings of the 43rd International ACM*

*SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, page 199–208, New York, NY, USA. Association for Computing Machinery.

[Liu et al., 2020b] Liu, B., Zhu, C., Li, G., Zhang, W., Lai, J., Tang, R., He, X., Li, Z., and Yu, Y. (2020b). AutoFIS: Automatic Feature Interaction Selection in Factorization Models for Click-Through Rate Prediction. In Gupta, R., Liu, Y., Tang, J., and Prakash, B. A., editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 2636–2645. ACM.

[Liu et al., 2023a] Liu, H., Li, B., Wu, H., Liang, H., Huang, Y., Li, Y., Ghanem, B., and Zheng, Y. (2023a). Combating mode collapse via offline manifold entropy estimation. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press.

[Liu et al., 2017] Liu, H., Zhu, M., Meng, X., Hu, Y., and Wang, H. (2017). Dual Based DSP Bidding Strategy and its Application. *CoRR*, abs/1705.09416.

[Liu et al., 2022] Liu, M., Hu, Z., Lai, Z., Zheng, D., and Nie, X. (2022). Real-time Bidding Strategy in Display Advertising: An Empirical Analysis. *CoRR*, abs/2212.02222.

[Liu et al., 2020c] Liu, M., Jiaxing, L., Hu, Z., Liu, J., and Nie, X. (2020c). A Dynamic Bidding Strategy Based on Model-Free Reinforcement Learning in Display Advertising. *IEEE Access*, 8:213587–213601.

[Liu et al., 2019] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692.

[Liu et al., 2014] Liu, Y., Xie, M., and Lakshmanan, L. V. S. (2014). Recommending user generated item lists. In Kobsa, A., Zhou, M. X., Ester, M., and Koren, Y., editors, *Eighth ACM Conference on Recommender Systems, RecSys '14, Foster City, Silicon Valley, CA, USA - October 06 - 10, 2014*, pages 185–192. ACM.

[Liu et al., 2023b] Liu, Z., Sun, L., Weng, C., Chen, Q., and Huo, C. (2023b). Gaussian Graph with Prototypical Contrastive Learning in E-Commerce Bundle Recommendation. *CoRR*, abs/2307.13468.

[Lowe et al., 2017] Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. (2017). Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6379–6390.

[Lyu et al., 2022] Lyu, F., Tang, X., Guo, H., Tang, R., He, X., Zhang, R., and Liu, X. (2022). Memorize, Factorize, or be Naive: Learning Optimal Feature Interaction Methods for CTR Prediction. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 1450–1462, Los Alamitos, CA, USA. IEEE Computer Society.

[Ma et al., 2024] Ma, Y., He, Y., Wang, X., Wei, Y., Du, X., Fu, Y., and Chua, T. (2024). MultiCBR: Multi-view Contrastive Learning for Bundle Recommendation. *ACM Trans. Inf. Syst.*, 42(4):100:1–100:23.

[Marchetti-Spaccamela and Vercellis, 1995] Marchetti-Spaccamela, A. and Vercellis, C. (1995). Stochastic on-line knapsack problems. *Math. Program.*, 68:73–104.

[Mentzer et al., 2023] Mentzer, F., Minnen, D., Agustsson, E., and Tschannen, M. (2023). Finite Scalar Quantization: VQ-VAE Made Simple. *CoRR*, abs/2309.15505.

[Miralles-Pechuán et al., 2023] Miralles-Pechuán, L., Qureshi, M. A., and Namee, B. M. (2023). Real-time bidding campaigns optimization using user profile settings. *Electron. Commer. Res.*, 23(2):1297–1322.

[Miralles-Pechuán et al., 2018] Miralles-Pechuán, L., Ponce, H., and Martínez-Villaseñor, L. (2018). A novel methodology for optimizing display advertising campaigns using genetic algorithms. *Electronic Commerce Research and Applications*, 27:39–51.

[Mirza and Osindero, 2014] Mirza, M. and Osindero, S. (2014). Conditional Generative Adversarial Nets. *CoRR*, abs/1411.1784.

[Mou et al., 2022] Mou, Z., Huo, Y., Bai, R., Xie, M., Yu, C., Xu, J., and Zheng, B. (2022). Sustainable Online Reinforcement Learning for Auto-bidding. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

[Myers et al., 2024] Myers, D., Mohawesh, R., Chellaboina, V. I., Sathvik, A. L., Venkatesh, P., Ho, Y.-H., Henshaw, H., Alhawawreh, M., Berdik, D., and Jararweh, Y. (2024). Foundation and large language models: fundamentals, challenges, opportunities, and social impacts. *Cluster Computing*, 27(1):1–26.

[Nichol et al., 2022] Nichol, A. Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. (2022). GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S., editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 16784–16804. PMLR.

[Nuara et al., 2022] Nuara, A., Trovò, F., Gatti, N., and Restelli, M. (2022). Online joint bid/daily budget optimization of Internet advertising campaigns. *Artif. Intell.*, 305:103663.

[Numberly, 2024] Numberly (2024). Real-Time Bidding (RTB) seen through the lens of machine learning research . https://numberly.com/en/real-time-bidding-rtb-seen-through-the-lens-of-machine-learning-research/. [Accessed 25-06-2024].

[OpenAI, 2023] OpenAI (2023). GPT-4 Technical Report. *CoRR*, abs/2303.08774.

[OpenAI, 2024] OpenAI (2024). Latency optimization Documentation. https://platform.openai.com/docs/guides/latency-optimization. [Accessed 25-06-2024].

[Pan et al., 2018] Pan, J., Xu, J., Ruiz, A. L., Zhao, W., Pan, S., Sun, Y., and Lu, Q. (2018). Field-weighted Factorization Machines for Click-Through Rate Prediction in Display Advertising. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1349–1357, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

[Park et al., 2019] Park, T., Liu, M., Wang, T., and Zhu, J. (2019). Semantic Image Synthesis With Spatially-Adaptive Normalization. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 2337–2346. Computer Vision Foundation / IEEE.

[Patashnik et al., 2021] Patashnik, O., Wu, Z., Shechtman, E., Cohen-Or, D., and Lischinski, D. (2021). StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 2065–2074. IEEE.

[Pathak et al., 2017] Pathak, A., Gupta, K., and McAuley, J. (2017). Generating and Personalizing Bundle Recommendations on Steam. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, page 1073–1076, New York, NY, USA. Association for Computing Machinery.

[Peebles and Xie, 2023] Peebles, W. and Xie, S. (2023). Scalable Diffusion Models with Transformers. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 4172–4182. IEEE.

[Perlich et al., 2012] Perlich, C., Dalessandro, B., Hook, R., Stitelman, O., Raeder, T., and Provost, F. (2012). Bid optimizing and inventory scoring in targeted online advertising. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, page 804–812, New York, NY, USA. Association for Computing Machinery.

[Pieters and Wiering, 2018] Pieters, M. and Wiering, M. A. (2018). Comparing Generative Adversarial Network Techniques for Image Creation and Modification. *CoRR*, abs/1803.09093.

[Qin et al., 2014] Qin, L., Chen, S., and Zhu, X. (2014). Contextual Combinatorial Bandit and its Application on Diversified Online Recommendation. In Zaki, M. J., Obradovic, Z., Tan, P., Banerjee, A., Kamath, C., and Parthasarathy, S., editors, *Proceedings of the 2014 SIAM International Conference on Data Mining, Philadelphia, Pennsylvania, USA, April 24-26, 2014*, pages 461–469. SIAM.

[Radford et al., 2021] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

[Radford et al., 2019] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI blog*.

[Ramesh et al., 2022] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. (2022). Hierarchical Text-Conditional Image Generation with CLIP Latents. *CoRR*, abs/2204.06125.

[Ramesh et al., 2021] Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. (2021). Zero-Shot Text-to-Image Generation. In Meila, M. and Zhang, T., editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR.

[Razavi et al., 2019] Razavi, A., van den Oord, A., and Vinyals, O. (2019). Generating Diverse High-Fidelity Images with VQ-VAE-2. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R., 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 14837–14847.

[Ren et al., 2019] Ren, K., Qin, J., Zheng, L., Yang, Z., Zhang, W., and Yu, Y. (2019). Deep Landscape Forecasting for Real-time Bidding Advertising. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19. ACM.

[Ren et al., 2018] Ren, K., Zhang, W., Chang, K., Rong, Y., Yu, Y., and Wang, J. (2018). Bidding Machine: Learning to Bid for Directly Optimizing Profits in Display Advertising. *IEEE Trans. Knowl. Data Eng.*, 30(4):645–659.

[Rendle, 2010] Rendle, S. (2010). Factorization Machines. In *2010 IEEE International Conference on Data Mining*, pages 995–1000.

[Rezende and Mohamed, 2015] Rezende, D. J. and Mohamed, S. (2015). Variational Inference with Normalizing Flows. In Bach, F. R. and Blei, D. M., 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 1530–1538. JMLR.org.

[Richardson et al., 2007] Richardson, M., Dominowska, E., and Ragno, R. (2007). Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, page 521–530, New York, NY, USA. Association for Computing Machinery.

[Roberts et al., 2018] Roberts, A., Engel, J. H., Raffel, C., Hawthorne, C., and Eck, D. (2018). A Hierarchical Latent Vector Model for Learning Long-Term Structure in Music. In Dy, J. G. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4361–4370. PMLR.

[Rombach et al., 2022a] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022a). High-Resolution Image Synthesis with Latent Diffusion Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE.

[Rombach et al., 2022b] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022b). High-Resolution Image Synthesis with Latent Diffusion Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE.

[Ruiz et al., 2023] Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. (2023). DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 22500–22510. IEEE.

[Saharia et al., 2022] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, S. K. S., Lopes, R. G., Ayan, B. K., Salimans, T., Ho, J., Fleet,

D. J., and Norouzi, M. (2022). Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022.*

[Sanner et al., 2023] Sanner, S., Balog, K., Radlinski, F., Wedin, B., and Dixon, L. (2023). Large Language Models are Competitive Near Cold-start Recommenders for Language- and Item-based Preferences. In Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T. D., Basilico, J., Pizzato, L., and Song, Y., editors, *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 890–896. ACM.

[Sar Shalom et al., 2016] Sar Shalom, O., Koenigstein, N., Paquet, U., and Vanchinathan, H. P. (2016). Beyond Collaborative Filtering: The List Recommendation Problem. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 63–72, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

[Schroff et al., 2015a] Schroff, F., Kalenichenko, D., and Philbin, J. (2015a). FaceNet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 815–823. IEEE Computer Society.

[Schroff et al., 2015b] Schroff, F., Kalenichenko, D., and Philbin, J. (2015b). FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823.

[SETUPAD, 2024] SETUPAD (2024). What is Real-Time Bidding (RTB)? `https://setupad.com/blog/real-time-bidding/`. [Accessed 26-06-2024].

[Shih et al., 2020] Shih, W., Lu, Y., Tsai, H., and Huang, J. (2020). An Expected Win Rate-Based Real Time Bidding Strategy for Branding Campaign by the Model-Free Reinforcement Learning Model. *IEEE Access*, 8:151952–151967.

[Shih et al., 2023] Shih, W.-Y., Lai, H.-C., and Huang, J.-L. (2023). A Robust Real Time Bidding Strategy Against Inaccurate CTR Predictions by Using Cluster Expected Win Rate. *IEEE Access*, 11:126917–126926.

[SmartyAds, 2017] SmartyAds (2017). 1st-Price Auction, 2nd-Price, and Header-Bidding. `https://smartyads.com/blog/smartyads-dual-auction-soft-transition-towards-first-price-auctions`. [Accessed 26-06-2024].

[Smith et al., 2023] Smith, J. T. H., Warrington, A., and Linderman, S. W. (2023). Simplified State Space Layers for Sequence Modeling. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

[Sohn, 2016] Sohn, K. (2016). Improved Deep Metric Learning with Multi-class N-pair Loss Objective. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1849–1857.

[Song et al., 2019] Song, W., Shi, C., Xiao, Z., Duan, Z., Xu, Y., Zhang, M., and Tang, J. (2019). AutoInt: Automatic Feature Interaction Learning via Self-Attentive Neural Networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19. ACM.

[Sun et al., 2019] Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., and Jiang, P. (2019). BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In Zhu, W., Tao, D., Cheng, X., Cui, P., Rundensteiner, E. A., Carmel, D., He, Q., and Yu, J. X., editors, *Proceedings of the 28th ACM International*

*Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 1441–1450. ACM.

[Sung et al., 2018] Sung, F., Yang, Y., Zhang, L., Xiang, T., Torr, P. H., and Hospedales, T. M. (2018). Learning to Compare: Relation Network for Few-Shot Learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1199–1208.

[Takida et al., 2022] Takida, Y., Liao, W.-H., Lai, C.-H., Uesaka, T., Takahashi, S., and Mitsufuji, Y. (2022). Preventing oversmoothing in VAE via generalized variance parameterization. *Neurocomputing*, 509:137–156.

[Tewel et al., 2023] Tewel, Y., Gal, R., Chechik, G., and Atzmon, Y. (2023). Key-Locked Rank One Editing for Text-to-Image Personalization. In Brunvand, E., Sheffer, A., and Wimmer, M., editors, *ACM SIGGRAPH 2023 Conference Proceedings, SIGGRAPH 2023, Los Angeles, CA, USA, August 6-10, 2023*, pages 12:1–12:11. ACM.

[Thomaidou et al., 2013] Thomaidou, S., Lourentzou, I., Katsivelis-Perakis, P., and Vazirgiannis, M. (2013). Automated snippet generation for online advertising. In He, Q., Iyengar, A., Nejdl, W., Pei, J., and Rastogi, R., editors, *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 1841–1844. ACM.

[Touvron et al., 2023] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Canton-Ferrer, C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. (2023). Llama 2: Open Foundation and Fine-Tuned Chat Models. *CoRR*, abs/2307.09288.

[Tumanyan et al., 2023] Tumanyan, N., Geyer, M., Bagon, S., and Dekel, T. (2023). Plug-and-Play Diffusion Features for Text-Driven Image-to-Image Translation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 1921–1930. IEEE.

[Vaddamanu et al., 2022] Vaddamanu, P., Aggarwal, V., Guda, B. P. R., Srinivasan, B. V., and Chhaya, N. (2022). Harmonized Banner Creation from Multimodal Design Assets. In Barbosa, S. D. J., Lampe, C., Appert, C., and Shamma, D. A., editors, *CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022 - 5 May 2022, Extended Abstracts*, pages 217:1–217:7. ACM.

[van den Oord et al., 2017] van den Oord, A., Vinyals, O., and Kavukcuoglu, K. (2017). Neural Discrete Representation Learning. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6306–6315.

[Vasile et al., 2017] Vasile, F., Lefortier, D., and Chapelle, O. (2017). Cost-sensitive Learning for Utility Optimization in Online Advertising Auctions. In *Proceedings of the ADKDD'17, Halifax, NS, Canada, August 13 - 17, 2017*, pages 8:1–8:6. ACM.

[Vaswani et al., 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is All you Need. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

[Vijaikumar et al., 2020] Vijaikumar, M., Shevade, S. K., and Murty, M. N. (2020). GRAM-SMOT: Top-N Personalized Bundle Recommendation via Graph Attention Mechanism and Submodular Optimization. In Hutter, F., Kersting, K., Lijffijt, J., and Valera, I., editors, *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2020, Ghent, Belgium, September 14-18, 2020, Proceedings, Part III*, volume 12459 of *Lecture Notes in Computer Science*, pages 297–313. Springer.

[Wang et al., 2023] Wang, F., Gu, H., Li, D., Lu, T., Zhang, P., and Gu, N. (2023). Towards Deeper, Lighter and Interpretable Cross Network for CTR Prediction. In Frommholz, I., Hopfgartner, F., Lee, M., Oakes, M., Lalmas, M., Zhang, M., and Santos, R. L. T., editors, *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 2523–2533. ACM.

[Wang et al., 2022a] Wang, F., Wang, Y., Li, D., Gu, H., Lu, T., Zhang, P., and Gu, N. (2022a). Enhancing CTR Prediction with Context-Aware Feature Representation Learning. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22. ACM.

[Wang et al., 2022b] Wang, H., Du, C., Fang, P., Yuan, S., He, X., Wang, L., and Zheng, B. (2022b). ROI-Constrained Bidding via Curriculum-Guided Bayesian Reinforcement Learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22. ACM.

[Wang et al., 2017a] Wang, J., Zhang, W., and Yuan, S. (2017a). Display Advertising with Real-Time Bidding (RTB) and Behavioural Targeting. *Found. Trends Inf. Retr.*, 11(4-5):297–435.

[Wang et al., 2017b] Wang, R., Fu, B., Fu, G., and Wang, M. (2017b). Deep & Cross Network for Ad Click Predictions. In *Proceedings of the ADKDD'17, Halifax, NS, Canada, August 13 - 17, 2017*, pages 12:1–12:7. ACM.

[Wang et al., 2021a] Wang, R., Shivanna, R., Cheng, D. Z., Jain, S., Lin, D., Hong, L., and Chi, E. H. (2021a). DCN V2: Improved Deep & Cross Network and Practical Lessons for Web-scale Learning to Rank Systems. In Leskovec, J., Grobelnik, M., Najork, M., Tang, J., and Zia, L., editors, *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 1785–1797. ACM / IW3C2.

[Wang et al., 2019] Wang, X., He, X., Wang, M., Feng, F., and Chua, T. (2019). Neural Graph Collaborative Filtering. In Piwowarski, B., Chevalier, M., Gaussier, É., Maarek, Y., Nie, J., and Scholer, F., editors, *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 165–174. ACM.

[Wang et al., 2017c] Wang, Y., Liu, J., Liu, Y., Hao, J., He, Y., Hu, J., Yan, W. P., and Li, M. (2017c). LADDER: A Human-Level Bidding Agent for Large-Scale Real-Time Online Auctions. *CoRR*, abs/1708.05565.

[Wang et al., 2021b] Wang, Z., She, Q., and Zhang, J. (2021b). MaskNet: Introducing Feature-Wise Multiplication to CTR Ranking Models by Instance-Guided Mask. In *Proceedings of the 3rd Workshop on Deep Learning Practice for High-Dimensional Sparse Data with KDD 2021.*

[Wei et al., 2022a] Wei, P., Liu, S., Yang, X., Wang, L., and Zheng, B. (2022a). Towards Personalized Bundle Creative Generation with Contrastive Non-Autoregressive Decoding. In Amigó, E., Castells, P., Gonzalo, J., Carterette, B., Culpepper, J. S., and Kazai, G., editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 2634–2638. ACM.

[Wei et al., 2022b] Wei, P., Yang, X., Liu, S., Wang, L., and Zheng, B. (2022b). CREATER: CTR-driven Advertising Text Generation with Controlled Pre-Training and Contrastive Fine-Tuning. In Loukina, A., Gangadharaiah, R., and Min, B., editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track, NAACL 2022, Hybrid: Seattle, Washington, USA + Online, July 10-15, 2022*, pages 9–17. Association for Computational Linguistics.

[Wei et al., 2023] Wei, Y., Liu, X., Ma, Y., Wang, X., Nie, L., and Chua, T. (2023). Strategy-aware Bundle Recommender System. In Chen, H., Duh, W. E., Huang, H., Kato, M. P., Mothe, J., and Poblete, B., editors, *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, pages 1198–1207. ACM.

[Wen et al., 2022] Wen, C., Xu, M., Zhang, Z., Zheng, Z., Wang, Y., Liu, X., Rong, Y., Xie, D., Tan, X., Yu, C., Xu, J., Wu, F., Chen, G., Zhu, X., and Zheng, B. (2022). A Cooperative-Competitive Multi-Agent Framework for Auto-bidding in Online Advertising. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22. ACM.

[Wermser et al., 2011a] Wermser, H., Rettinger, A., and Tresp, V. (2011a). Modeling and Learning Context-Aware Recommendation Scenarios Using Tensor Decomposition. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 137–144.

[Wermser et al., 2011b] Wermser, H., Rettinger, A., and Tresp, V. (2011b). Modeling and Learning Context-Aware Recommendation Scenarios Using Tensor Decomposition. In *International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2011, Kaohsiung, Taiwan, 25-27 July 2011*, pages 137–144. IEEE Computer Society.

[Wu et al., 2018a] Wu, D., Chen, X., Yang, X., Wang, H., Tan, Q., Zhang, X., Xu, J., and Gai, K. (2018a). Budget Constrained Bidding by Model-free Reinforcement Learning in Display Advertising. In *Proceedings of the 27th ACM International Con-*

*ference on Information and Knowledge Management*, CIKM '18, page 1443–1451, New York, NY, USA. Association for Computing Machinery.

[Wu et al., 2021] Wu, H., Xu, J., Wang, J., and Long, M. (2021). Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 22419–22430.

[Wu et al., 2018b] Wu, W., Yeh, M.-Y., and Chen, M.-S. (2018b). Deep Censored Learning of the Winning Price in the Real Time Bidding. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 2526–2535, New York, NY, USA. Association for Computing Machinery.

[Wu et al., 2015] Wu, W. C.-H., Yeh, M.-Y., and Chen, M.-S. (2015). Predicting Winning Price in Real Time Bidding with Censored Data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, page 1305–1314, New York, NY, USA. Association for Computing Machinery.

[Xi et al., 2021] Xi, D., Chen, Z., Yan, P., Zhang, Y., Zhu, Y., Zhuang, F., and Chen, Y. (2021). Modeling the Sequential Dependence among Audience Multi-step Conversions with Multi-task Learning in Targeted Display Advertising. In Zhu, F., Ooi, B. C., and Miao, C., editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 3745–3755. ACM.

[Xiao et al., 2017] Xiao, J., Ye, H., He, X., Zhang, H., Wu, F., and Chua, T. (2017). Attentional Factorization Machines: Learning the Weight of Feature Interactions via Attention Networks. In Sierra, C., editor, *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 3119–3125. ijcai.org.

[Xie et al., 2010] Xie, M., Lakshmanan, L. V., and Wood, P. T. (2010). Breaking out of the box of recommendations: from items to packages. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, page 151–158, New York, NY, USA. Association for Computing Machinery.

[Xu et al., 2015] Xu, J., Lee, K.-c., Li, W., Qi, H., and Lu, Q. (2015). Smart Pacing for Effective Online Ad Campaign Optimization. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15. ACM.

[Xue et al., 2017] Xue, H.-J., Dai, X.-Y., Zhang, J., Huang, S., and Chen, J. (2017). Deep matrix factorization models for recommender systems. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 3203–3209. AAAI Press.

[Yang et al., 2019] Yang, X., Li, Y., Wang, H., Wu, D., Tan, Q., Xu, J., and Gai, K. (2019). Bid Optimization by Multivariable Control in Display Advertising. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 1966–1974, New York, NY, USA. Association for Computing Machinery.

[Yang and Zhai, 2022] Yang, Y. and Zhai, P. (2022). Click-through rate prediction in online advertising: A literature review. *Inf. Process. Manag.*, 59(2):102853.

[Yao et al., 2023] Yao, S., Pan, X., Dong, H., Kong, C., and Wu, X. (2023). Adaptive-Bert Network for Advertising Text Generation. In *2023 IEEE 6th International Conference on Electronic Information and Communication Technology (ICEICT)*, pages 192–196.

[Yin et al., 2023] Yin, B., Xie, J., Qin, Y., Ding, Z., Feng, Z., Li, X., and Lin, W. (2023). Heterogeneous Knowledge Fusion: A Novel Approach for Personalized Recommendation via LLM. In Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T. D., Basilico, J., Pizzato, L., and Song, Y., editors, *Proceedings of the 17th ACM Conference on*

*Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 599–601. ACM.

[Ying et al., 2018] Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., and Leskovec, J. (2018). Graph Convolutional Neural Networks for Web-Scale Recommender Systems. In Guo, Y. and Farooq, F., editors, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 974–983. ACM.

[Ying and Zhizhong, 2009] Ying, S. and Zhizhong, M. (2009). A Genetic Algorithm Based Approach For Mixed-Media Advertising Budgeting. In *2009 WRI Global Congress on Intelligent Systems*, volume 3, pages 269–273.

[Yu et al., 2020] Yu, F., Liu, Z., Liu, Q., Zhang, H., Wu, S., and Wang, L. (2020). Deep Interaction Machine: A Simple but Effective Model for High-order Feature Interactions. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 2285–2288, New York, NY, USA. Association for Computing Machinery.

[Yu et al., 2022] Yu, J., Li, X., Koh, J. Y., Zhang, H., Pang, R., Qin, J., Ku, A., Xu, Y., Baldridge, J., and Wu, Y. (2022). Vector-quantized Image Modeling with Improved VQGAN. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

[Zhang et al., 2021a] Zhang, C., Barbano, R., and Jin, B. (2021a). Conditional Variational Autoencoder for Learned Image Reconstruction. *Comput.*, 9(11):114.

[Zhang et al., 2017] Zhang, H., Zhang, W., Rong, Y., Ren, K., Li, W., and Wang, J. (2017). Managing Risk of Bidding in Display Advertising. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM 2017. ACM.

[Zhang et al., 2023a] Zhang, J., Bao, K., Zhang, Y., Wang, W., Feng, F., and He, X. (2023a). Is ChatGPT Fair for Recommendation? Evaluating Fairness in Large

Language Model Recommendation. In Zhang, J., Chen, L., Berkovsky, S., Zhang, M., Noia, T. D., Basilico, J., Pizzato, L., and Song, Y., editors, *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 993–999. ACM.

[Zhang and Zhang, 2023] Zhang, P. and Zhang, J. (2023). MemoNet: Memorizing All Cross Features' Representations Efficiently via Multi-Hash Codebook Network for CTR Prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23. ACM.

[Zhang et al., 2023b] Zhang, P., Zheng, Z., and Zhang, J. (2023b). FiBiNet++: Reducing Model Size by Low Rank Feature Interaction Layer for CTR Prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23. ACM.

[Zhang et al., 2021b] Zhang, W., Kitts, B., Han, Y., Zhou, Z., Mao, T., He, H., Pan, S., Flores, A., Gultekin, S., and Weissman, T. (2021b). MEOW: A Space-Efficient Nonparametric Bid Shading Algorithm. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 3928–3936, New York, NY, USA. Association for Computing Machinery.

[Zhang et al., 2014] Zhang, W., Yuan, S., and Wang, J. (2014). Optimal real-time bidding for display advertising. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 1077–1086, New York, NY, USA. Association for Computing Machinery.

[Zhang et al., 2022] Zhang, Z., Du, B., and Tong, H. (2022). SuGeR: A Subgraph-based Graph Convolutional Network Method for Bundle Recommendation. In Hasan, M. A. and Xiong, L., editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 4712–4716. ACM.

[Zhao et al., 2018] Zhao, J., Qiu, G., Guan, Z., Zhao, W., and He, X. (2018). Deep Reinforcement Learning for Sponsored Search Real-time Bidding. In Guo, Y. and Farooq, F., editors, *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 1021–1030. ACM.

[Zhou et al., 2022] Zhou, H., Yang, C., Gao, X., Chen, Q., Liu, G., and Chen, G. (2022). Multi-Objective Actor-Critics for Real-Time Bidding in Display Advertising. In Amini, M., Canu, S., Fischer, A., Guns, T., Novak, P. K., and Tsoumakas, G., editors, *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2022, Grenoble, France, September 19-23, 2022, Proceedings, Part IV*, volume 13716 of *Lecture Notes in Computer Science*, pages 20–37. Springer.

[Zhou et al., 2021] Zhou, T., He, H., Pan, S., Karlsson, N., Shetty, B., Kitts, B., Gligorijevic, D., Gultekin, S., Mao, T., Pan, J., Zhang, J., and Flores, A. (2021). An Efficient Deep Distribution Network for Bid Shading in First-Price Auctions. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21. ACM.

[Zhu et al., 2023] Zhu, S., Cui, C., Hu, J., Shen, Q., Ji, Y., and Wei, Z. (2023). Text2Bundle: Towards Personalized Query-based Bundle Generation. *CoRR*, abs/2310.18004.

[Zhu et al., 2014] Zhu, T., Harrington, P., Li, J., and Tang, L. (2014). Bundle recommendation in ecommerce. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '14, page 657–666, New York, NY, USA. Association for Computing Machinery.

[Zhu et al., 2017] Zhu, W., Shih, W., Lee, Y., Peng, W., and Huang, J. (2017). A gamma-based regression for winning price estimation in real-time bidding advertising. In Nie, J., Obradovic, Z., Suzumura, T., Ghosh, R., Nambiar, R., Wang, C., Zang, H., Baeza-Yates, R., Hu, X., Kepner, J., Cuzzocrea, A., Tang, J., and Toyoda, M., editors, *2017 IEEE International Conference on Big Data (IEEE BigData 2017), Boston, MA, USA, December 11-14, 2017*, pages 1610–1619. IEEE Computer Society.