Extreme Meta-Classification for Large-Scale Zero-Shot Retrieval

Sachin Yadav*†‡
t-sacyadav@microsoft.com
Microsoft Research
India

Deepak Saini*
desaini@microsoft.com
Microsoft
USA

Anirudh Buvanesh*
t-abuvanesh@microsoft.com
Microsoft Research
India

Bhawna Paliwal
bhawna@microsoft.com
Microsoft Research
India

Kunal Dahiya
kunalsdahiya@gmail.com
IIT Delhi
India

Siddarth Asokan‡
sasokan@microsoft.com
Microsoft Research
India

Yashoteja Prabhu
yprabhu@microsoft.com
Microsoft Research
India

Jian Jiao
Jian.Jiao@microsoft.com
Microsoft
USA

Manik Varma
manik@microsoft.com
Microsoft Research
India

ABSTRACT

We develop accurate and efficient solutions for large-scale retrieval tasks where novel (*zero-shot*) items can arrive continuously at a rapid pace. Conventional Siamese-style approaches embed both queries and items through a small encoder and retrieve the items lying closest to the query. While this approach allows efficient addition and retrieval of novel items, the small encoder lacks sufficient capacity for the necessary world knowledge in complex retrieval tasks. The extreme classification approaches have addressed this by learning a separate classifier for each item observed in the training set which significantly increases the representation capacity of the model. Such classifiers outperform Siamese approaches on observed items, but cannot be trained for novel items due to data and latency constraints. To bridge these gaps, this paper develops: (1) A **new algorithmic framework**, EMMETT, which efficiently synthesizes classifiers on-the-fly for novel items, by relying on the readily available classifiers for observed items; (2) A **new algorithm**, IRENE, which is a simple and effective instance of EMMETT that is specifically suited for large-scale deployments, and (3) A **new theoretical framework** for analyzing the generalization performance in large-scale zero-shot retrieval which guides our algorithm and training related design decisions.

Comprehensive experiments are conducted on a wide range of retrieval tasks which demonstrate that IRENE improves the zero-shot retrieval accuracy by up to 15% points in Recall@10 when added on top of leading encoders. Additionally, on an online A/B test in a large-scale ad retrieval task in a major search engine, IRENE improved the ad click-through rate by 4.2%. Lastly, we validate our design choices through extensive ablative experiments. The source code for IRENE is available at https://aka.ms/irene.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning by classification**.

KEYWORDS

extreme classification, large-scale retrieval, zero-shot retrieval, sponsored search advertising

ACM Reference Format:
Sachin Yadav, Deepak Saini, Anirudh Buvanesh, Bhawna Paliwal, Kunal Dahiya, Siddarth Asokan, Yashoteja Prabhu, Jian Jiao, and Manik Varma. 2024. Extreme Meta-Classification for Large-Scale Zero-Shot Retrieval. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3637528.3672046

1 INTRODUCTION

Large-scale retrieval involves retrieving the items relevant to a query from a pool of hundreds of millions of candidate items. Such tasks frequently arise in modern-day web applications such as web search [6], computational advertising [3], product recommendation [10], and so on. To optimize user satisfaction, the retrieved results need to be highly relevant to the query and delivered in real-time, typically within milliseconds. Additionally, due to the exponential growth of digital content, novel items get introduced into these systems in vast quantities daily, which need to be swiftly processed and inserted into the candidate pool to ensure up-to-date results. This paper aims to develop highly accurate and efficient solutions for problems of large-scale text-based retrieval with *novel* items (also referred to as *zero-shot* items). Note that this scenario is different from the one considered in [16, 46] where the task itself is zero-shot.

A widely used technique for large-scale retrieval is dense retrieval [52] where both queries and items are represented as embeddings in a shared low-dimensional space such that the items relevant to a query are positioned closer to it than the irrelevant

*Equal technical contribution. †Sachin Yadav led the project.
‡Corresponding authors are S. Yadav (sachinyadav7024@gmail.com) and S. Asokan (sasokan@microsoft.com)

KDD '24, August 25–29, 2024, Barcelona, Spain

ACM ISBN 979-8-4007-0490-1/24/08
https://doi.org/10.1145/3637528.3672046

ones. The relevant items, typically few in number, are then retrieved in almost real-time using scalable Approximate Nearest Neighbour Search (ANNS) indices [41]. The accuracy of retrieved results depends on the quality of the derived query and item representations.

Typical dense retrievers are based on a Siamese encoder architecture which uses a common deep neural encoder to derive both query and item representations from their raw text inputs [23, 47]. Usually, a small and efficient encoder is utilized to reduce the representation latencies, which enables large-scale deployments and quick insertion of novel items. However, a small encoder often lacks the capacity to model the complexity inherent in retrieval tasks [15]. For instance, item descriptions frequently contain named entities, numbers, model numbers, and ambiguous phrases with issues of synonymy and polysemy whose resolution requires extensive world knowledge that cannot be contained within a small encoder [44]. As a result, the representations from these approaches are inferior and degrade the retrieval performance.

Recently, Extreme Classification (XC) methods have emerged as promising alternatives for large scale retrieval. Leading extreme classifiers augment a small Siamese encoder with a massive linear classifier layer at its output which significantly boosts the model capacity [12, 13]. Each item *observed* in the training set is endowed with its own classifier, which absorbs the world knowledge pertinent to the item when trained from the historical click logs. During retrieval, a query is first passed through the encoder and then projected onto its relevant items by applying classifiers. When there are enough clicked query samples for training, the classifier-based item representations can be more precise than the text-restricted representations from a small encoder. However, zero-shot retrieval is not supported, as classifiers cannot be trained for novel items with no clicked samples. Even if we could, learning new classifiers from scratch is time-consuming and delays the representation of novel items, making the candidate pool outdated.

This paper addresses the limitations of the existing approaches on large-scale zero-shot retrieval. Our primary research question is: ***How to construct accurate representations for novel items without significant computational overhead in large-scale retrieval tasks?***

We propose a novel algorithmic framework, ExtreMe METa-classificaTion (EMMETT), as an answer to this question. EMMETT extends conventional Extreme Classification to zero-short scenarios. Often in large-scale retrieval tasks, millions of observed items with associated user-clicked queries are available for training from historical logs. This leads to two key insights that underpin EMMETT's design:

First, ignoring any significant shifts in item distribution, extreme classifiers trained for a million observed items are likely to contain most of the world knowledge required for any novel item in a distilled and readily usable form. With this intuition, EMMETT builds a pipeline with two main modules: (1) A classifier selector ($\mathcal{S}$) module, which takes a novel item as input and rapidly shortlists a few observed item classifiers that are most informative for it, by using efficient filtering techniques, and (2) A meta-classifier generator ($\mathcal{G}$) module, which combines these shortlisted classifiers to synthesize a novel item's classifier with minimal latency. We call such a classifier derived from other classifiers as a *meta-classifier*. Second, a million observed items can also serve as a large number

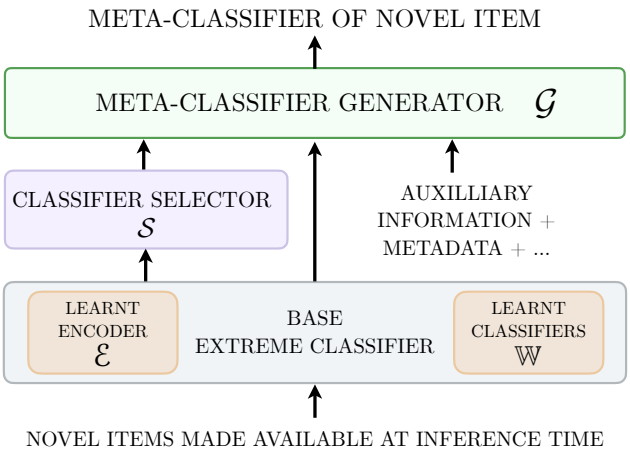

Figure 1: An overview of our proposed ExtreMe METa-classificaTion (EMMETT) framework. Given an extreme classification base (encoder $\mathcal{E}$ and classifiers $\mathbb{W}$), EMMETT consists of two modules, where (a) the classifier selector $\mathcal{S}$ retrieves the *most informative* classifiers for a novel item, and (b) the meta-classifier generator $\mathcal{G}$ combines the selected classifiers and other meta-data to create the meta-classifier.

of samples for optimally training the EMMETT modules. With a careful loss design, each observed item can be treated as a proxy for a zero-shot item which yields a massive training set for zero-shot retrieval. Training EMMETT on such a dataset can ensure robust generalization on novel items. We theoretically and empirically validate this claim in Sections **5** and **7**, respectively.

Figure **1** presents the architectural overview of EMMETT. EMMETT is a generic framework where different choices for the two modules and the training strategy can potentially yield algorithms with varying accuracy and efficiency trade-offs, which also presents ample opportunities for future research explorations.

In this paper, we develop one such algorithmic instantiation of the EMMETT framework, named IRENE, for Improved REtrieval of Novel itEms. Given a generic Siamese encoder augmented with extreme classifiers on observed items, IRENE is designed to add meta-classifier functionality to this base with minimal effort. The IRENE selects the classifiers associated with observed items by means of an efficient ANNS-search. The item representations are derived from the base Siamese encoder itself. This avoids training an additional module thus reducing the training and deployment efforts, but can also add noisy classifiers to the shortlist. To effectively filter the noise, combine the useful knowledge and synthesize meta-classifiers, IRENE trains a separate meta-classifier generator model that is based on a transformer architecture. Given the shortlist of the selected classifiers, the meta-classifier synthesis is highly efficient, requiring a forward pass over the single-layer transformer (cf. Section 7). IRENE training freezes the Siamese encoder and extreme classifiers, and only trains the generator using a weighted one-vs-all classification loss. Both novel item representation and retrieval are real-time and require few milliseconds in IRENE. Figure **2** depicts the IRENE architecture.

Theoretical analysis is a crucial tool which can guide the design of robust and effective machine learning systems. For this purpose, we develop a novel theoretical framework for analyzing the generalization performance in large-scale zero-shot retrieval. It is based on a key insight that, given a dense retriever that outputs calibrated query-item relevance scores, the zero-shot retrieval task is equivalent to a binary classification task with a query-item pair as our data sample. This equivalence allows us to leverage and further extend the theoretical ideas from traditional binary classification literature. Our results in Section **5** show that (1) One-vs-All loss in IRENE leads to strong zero-shot generalization performance by reducing the variance in the training loss, (2) Freezing extreme classifiers during IRENE avoids overfitting and is therefore crucial for robust zero-shot generalization, (3) Number of observed items and the Generator's complexity also affect the zero-shot generalization. These results are also validated empirically in Section **7**.

In summary, this paper makes the following key contributions:

(1) A **novel and generic algorithmic framework**, EMMETT, for learning accurate representations of novel items, thus improving the zero-shot retrieval performance.
(2) A **novel and efficient algorithm**, IRENE, that can be added to any Siamese encoder and is well-suited for large-scale real-world applications.
(3) A **novel theoretical framework** for analyzing the generalization performance in large-scale zero-shot retrieval.
(4) **Comprehensive experimental validation** demonstrating that IRENE can offer up to 15% gains in terms of *Recall*@10 with minimal overheads, in zero-shot retrieval on diverse (varying in their scale and application) benchmark datasets.
(5) **Online A/B tests** on a large-scale *Sponsored Search* application in a major search engine which shows that IRENE can improve ad click rate by 4% thereby demonstrating its real-world utility.

The source code for IRENE in available at https://aka.ms/irene. The project page is https://aka.ms/emmett.

2 RELATED WORK

This section provides a comprehensive review of the past literature that is pertinant to our work. It presents the different classes of approaches available for retrieval and highlights their scalability, accuracy and zero-shot capabilities.

Traditional statistical approaches: Traditional approaches to large-scale retrieval, including TF-IDF [22] and BM25 [34], relied on measuring the co-occurrences of terms in a query text and an item text. These methods utilized inverted term indices for efficient item search which scaled well and allowed rapid insertion of novel items. However, they lacked the ability to understand the context of the inputs. ZestXML [17], a recent method for large-scale zero-shot retrieval, extended the term-matching to also account for semantic correlations between the terms which improved the retrieval performance. Nevertheless, these traditional methods have been largely superseded by semantic matching techniques using deep networks, which offer superior accuracy.

Siamese-encoder approaches: These approaches employ deep neural encoders to embed both query and item texts into a shared, low-dimensional representation space. Leading approaches such as DPR [23], ANCE [47], MACLR [48], RocketQA [33] and NGAME [13], utilize a transformer-based encoder. ANCE, RocketQA and NGAME also implement advanced negative-mining strategies for robust encoder training. MACLR leverages self-supervised learning based on an inverse cloze task for pre-training. In these approaches, incorporating a novel item is efficient and requires just a single encoder pass over their raw features. However, to maintain real-time response rates, these methods are constrained to use a small transformer, which degrades retrieval accuracy due to lack of world knowledge necessary for large-scale retrieval. Recent works like ColBERT [26] and Semsup-XC [2] modify the Siamese architecture to permit token-level interactions between the query and item texts which improves retrieval accuracy. However, this enhancement also increases inference latency, making these models less suitable for practical deployments. Semsup-XC also relies on web-scraped meta-data to boost zero-shot performance, which can be expensive.

Extreme classification approaches: These approaches learn one-vs-all classifiers to represent each observed item in the training set [14, 15, 18, 19, 25, 29, 31, 37, 51]. These are highly accurate, offer low inference latencies and have been successfully applied to a wide range of large-scale retrieval tasks including document tagging [5], product-to-product recommendation [21, 24], and sponsored search [12, 13]. Unfortunately, most of these approaches do not have support for zero-shot items.

Among these, the most relevant to our work is DEXA [15]. DEXA learns a small set of classifiers, one classifier per cluster of items to improve the retrieval performance. DEXA can be viewed as a naive version of EMMETT with an inferior architecture and training loss, and is significantly outperformed by IRENE (see Section **6**).

Other zero-shot retrieval approaches: Several approaches have been proposed for zero-shot multi-label retrieval such as ESZSL [35], COSTA [30], LAF [28] *etc.* These works consider a few thousand labels (items) and typically do not scale to million items.

Over the past year, large language model based zero-shot retrieval is also gaining popularity [38]. These models are expensive and are not yet demonstrated for large-scale retrieval.

The works in [16, 46] study a different zero-shot setting where the task itself is novel with no available relevance signals for model training. They propose techniques based on domain transfer as solutions. Unlike these works, our setting assumes that abundant click data is available as relevance signals and focusses on leveraging them to improve the retrieval with zero-shot items.

3 EXTREME META-CLASSIFICATION

In this section, we introduce **EMMETT**, our proposed ExtreMe METa-classificaTion framework for improving zero-shot retrieval by means of incorporating the knowledge present in the learnt classifiers of observed items.

3.1 Preliminaries

We describe the notation and background that is necessary for the rest of this section. Consider queries X drawn from the space $\mathcal{X}$, which are mapped to items Z drawn from the item space $\mathcal{Z}$. During training, we have access to N data points $\{X_i\}_{i=1}^{N}$ and L *observed* items $\{Z_\ell\}_{\ell=1}^{L}$. Each data-item pair is associated with a label $y_{i\ell} = \{0, 1\}$, which is 1 when the item is relevant to the input query, and 0 when the item is irrelevant to the query.

Dense retrieval (DR) algorithms consider a deep feature encoder $\mathcal{E}$ that projects the data samples and items to the space of d-dimensional encoder representations, denoted as $\boldsymbol{x} = \mathcal{E}(X)$ and $\boldsymbol{z} = \mathcal{E}(Z)$, respectively, both $\mathcal{E}(\mathcal{Z}), \mathcal{E}(\mathcal{X})$ belonging to $\mathbb{R}^d$. The query and observed-item sets in this encoder space are represented as $\mathbb{X} = \{\boldsymbol{x}_i\}_{i=1}^N$, and $\mathbb{Z} = \{\boldsymbol{z}_\ell\}_{\ell=1}^L$, respectively. In Siamese DR algorithms, both query and item representations share the same encoder space. Items are retrieved for a given query, typically using approximate nearest neighbor search (ANNS), based on the similarity score $\boldsymbol{x}_i^\top \boldsymbol{z}_\ell$.
Extreme classification (XC) algorithms are built on the intuition that classifier-based item representations can surpass the performance of encoder-representations. In XC, each item $\boldsymbol{z}_\ell$ is represented by a 1-vs-all classifier $\boldsymbol{w}_\ell$, collectively denoted as $\mathbb{W} = \{\boldsymbol{w}_\ell\}_{\ell=1}^L$. These are trained using triplet-based [13] or cross-entropy-based [20] losses atop the encoder representation. The similarity score is calculated between the query representation and the classifiers ($\boldsymbol{x}_i^\top \boldsymbol{w}_\ell$). Thus, the XC module comprises an encoder and a set of learned classifiers $(\mathcal{E}, \mathbb{W})$. While effective in retrieving observed items, these methods face challenges in retrieving novel items.

3.2 The EMMETT Framework

We propose EMMETT, an **extreme meta-classification** framework for zero-shot generalization in the retrieval setting. The EMMETT framework is designed to develop highly accurate and efficient retrieval models capable of handling novel item retrieval during inference. Existing XC models, despite their accuracy through classifier-based approaches built atop encoder representations, suffer from high data and resource requirement and therefore, and do not generalize to the zero-shot setting.

The EMMETT framework considers training data $\mathbb{X}$ and $\mathbb{Z}$, a **base extreme classifier** $(\mathcal{E}, \mathbb{W})$, comprising the encoder $\mathcal{E}$ and observed-item classifiers $\mathbb{W}$, and a set of novel items $\mathbb{Z}_n = \{\boldsymbol{z}_\ell\}_{\ell=1}^{L_n}$ made available at inference time, EMMETT includes two modules:

- **Classifier Selector** $\mathcal{S}$: This module selects the *most informative* set of classifiers from the base XC model for a novel item $\boldsymbol{z} \in \mathbb{Z}_n$ introduced at inference time.
- **Meta-classifier Generator** $\mathcal{G}$: This module combines the classifiers selected by $\mathcal{S}$, and other auxiliary information (such as the encoder representation $\boldsymbol{z}$) to create the meta classifier $\boldsymbol{u}$ associated with item $\boldsymbol{z}$.

EMMETT offers flexibility in its implementation through a variety of design choices, such as the, base XC framework, choice of encoder architecture and training methods. The classifier selector's $\mathcal{S}$ design involves choosing a selection algorithm and determining the number of classifiers to select. An ideal selection algorithm should choose *informative* classifiers, taking into account factors such as named entities, numbers, ambiguous phrases, synonyms, etc. Since the task of classifier selection can be viewed as retrieval, any existing retrieval methods can be used. In the context of EMMETT we prioritize efficient classifier selection and quick incorporation of novel items. The generator's $\mathcal{G}$ design must take into consideration model complexity (which we discuss in Section 5), ensuring it effectively incorporates the selected classifiers and potential auxiliary data to learn the meta-classifier $\boldsymbol{u}$ without overfitting. Another component of EMMETT is in designing loss functions for towards zero-shot generalization, and choosing appropriate training strategies, such as end-to-end, modular, etc.

Figure 1 provides a visual illustration of these three components. EMMETT enables interpreting existing XC models' zero-shot generalizability. For example, in DEXA [15], given a novel item, its classifier can be selected based on cluster assignment, and then summed with its encoder representation. Similarly, in NGAME [13], the classifier selector is an indicator function, which returns a classifier only for the observed items. NGAME's decision tree, along with its encoder representation and classifier, can be viewed as its generator block. DEXA's and NGAME's relatively poorer zero-shot generalization can be attributed to their simplistic $\mathcal{S}$ and $\mathcal{G}$ architectures. An additional reason why these models perform poorly on novel items can traced back to their loss functions, which are not tailored for zero-shot performance.

In Section 4, we present IRENE, a specific instance of EMMETT designed for zero-shot generalization. This includes a thoughtful design of classifier selector and generator modules, considering training strategies, deployment ease, new item incorporation at inference, and loss functions prioritizing zero-shot performance. Theoretical analysis of IRENE's effectiveness is detailed in Section 5.

4 THE IRENE EXTREME META-CLASSIFIER

We now present the **IRENE** algorithm, our proposed approach for Improved REtrieval of Novel itEms with extreme meta-classification. IRENE is adaptable to any XC framework. Specifically, we utilize a 6-layer DistilBERT encoder, trained with state-of-the-art, computationally efficient algorithms such as NGAME [13], ANCE [47], MACLR [48], and DPR [23]. Given a real-world application such as product-to-product recommendations, document tagging, or matching user queries to advertiser keywords appropriate feature encoders may be chosen. After training, the encoder $\mathcal{E}$ and the observed-item classifiers $\mathbb{W}$ remain fixed through subsequent stages. The learnt classifiers are accessible via a lookup function C_{lf} based on the items' encoder representations, *i.e.,* $\boldsymbol{w}_\ell = C_{lf}(\boldsymbol{z}_\ell)$.
The classifier selector $\mathcal{S}$, for any item $\boldsymbol{z}_\ell$, either observed or novel, retrieves K classifiers of related items using an ANNS index built on the item representations $\mathcal{E}(\mathbb{Z}_s)$, served via approaches such as DiskANN [41]. The selection is based on a maximum inner product search (MIPS) between the novel item, and observed items $\arg\max_K \{\boldsymbol{z}_\ell^\top \boldsymbol{z}_o;\ \forall\, \boldsymbol{z}_o \in \mathbb{Z}\}$. The complexity of these approaches has typically been shown to be logarithmic in the number of items, *i.e.,* $O(\log L)$. The choice of K, a key hyper-parameter, balances model capacity and complexity, which we formalize in Section 5 and validate in Section 7. Empirically, we found $K \approx 3$ to work well.
The meta-classifier generator $\mathcal{G}_{\boldsymbol{\phi}}$ is based on a transformer architecture, inspired by their capability to be universal approximators [50] and their success in learning embeddings in few-shot scenarios [49]. While the transformer layers are capable of handling a variety of input embeddings, in IRENE, the input sequence consists of the selected classifiers and the novel item's encoder representation and is passed through self-attention and linear blocks. In particular, the meta classifier is given by $\boldsymbol{u}_\ell = \mathcal{G}_{\boldsymbol{\phi}}(\boldsymbol{z}_\ell, \mathcal{S}(\boldsymbol{z}_\ell))$ and each layer can be expressed as as:

$$\text{Linear}(\text{Self-Attention}(\boldsymbol{z}_\ell + \boldsymbol{t}_{enc}, \boldsymbol{w}_\ell^1 + \boldsymbol{t}_{clf}, \boldsymbol{w}_\ell^2 + \boldsymbol{t}_{clf}, \ldots), \quad (1)$$

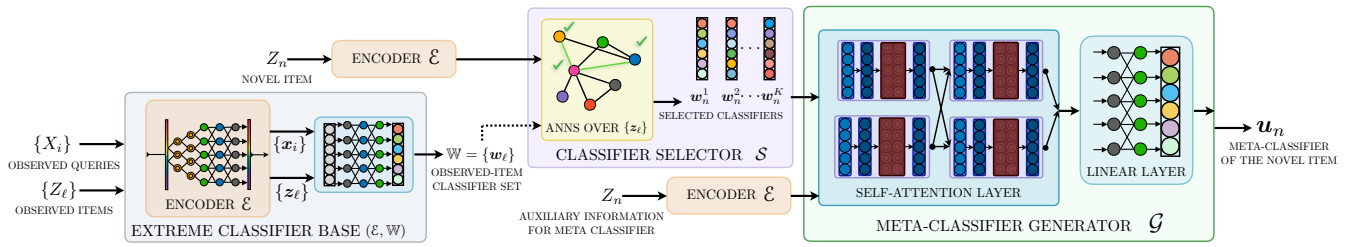

Figure 2: The IRENE extreme meta-classifier. IRENE comprises (a) A base extreme classifier encoder and the classifiers trained via a standard algorithm; (b) A classifier selector $\mathcal{S}$, which, given the encoder representation of a novel item Z_n, retrieves K classifiers based on an approximate nearest neighbor search (ANNS), and (c) A transformer-based meta-classifier generator $\mathcal{G}$. The meta-classifier u_n of Z_n is computed as given in Equation (1).

where $\boldsymbol{w}_\ell \in \mathcal{S}(z_\ell)$, t_{enc} and t_{clf} are learnable *type embeddings* that distinguish between encoder and classifier inputs. The transformer layer is capable of learning correlations between the classifiers and the encoder representation, yields superior meta-classifiers over methods that consider naive weighted summation for the generator (cf. Section 7). The computational complexity is $O\left((K+1)^2 d^2\right)$ for the self attention layer [42], and $O\left((K+1)\, d\right)$ for the linear layer. Figure 2 illustrates the modules present in the IRENE framework.

4.1 IRENE: Training and Inference

In this section, we outline the training strategy for IRENE. Firstly, we **assume that the representation encoder $\mathcal{E}$, and the set of observed-item classifiers $\mathbb{W} = \{\boldsymbol{w}_\ell\}_{\ell=1}^{L}$ are made available to us apriori, and remain fixed.** With these components $(\mathcal{E}, \mathbb{W})$, the ANNS index is constructed over the encoder representations of the observed items $\mathbb{Z}$. Classifier selection is as described in Section 4.

IRENE's meta-classifier training employs a binary cross-entropy loss reformulated for zero-shot performance. In particular, to simulate novel items at training time, given an observed item $\boldsymbol{z}_\ell$, instead of using the item's own classifier $\boldsymbol{w}_\ell$, the meta-classifier $\boldsymbol{u}_\ell$, derived from the selected classifiers, is utilized. That is, an observed item its trained with its selected *neighboring* classifiers and not its own. The original targets $y_{i\ell}$ remain unchanged. The loss is given by:

$$\mathcal{L}_{\mathcal{G}} = \sum_{\boldsymbol{x}_i \in \mathbb{X}} \sum_{\boldsymbol{z}_\ell \in \mathbb{Z}} \left(C y_{i\ell}\, ln\left(\sigma(\boldsymbol{x}_i^\top \boldsymbol{u}_\ell)\right) + (1 - y_{i\ell}) \ln\left(1 - \sigma(\boldsymbol{x}_i^\top \boldsymbol{u}_\ell)\right)\right) \quad (2)$$

where $\sigma(\cdot)$ is the sigmoid function, $\boldsymbol{u}_\ell = \mathcal{G}_{\boldsymbol{\phi}}(\boldsymbol{z}_\ell, \mathcal{S}(\boldsymbol{z}_\ell))$, $\boldsymbol{x} = \mathcal{E}(X)$, and C is the weight for misclassified positive query-item pairs. This weight addresses the imbalance between positive and negative query-item pairs in standard XC settings, as misclassifying positives significantly impacts performance. [9]. We link the likelihood of encountering positive-pairs to zero-shot generalization performance in Section 5.

To facilitate training, the loss is approximated using standard negative mining strategies, with the positive, and negative-mined items associated with a query represented by $\mathbb{Z}_{o,i}^{+}$ and $\mathbb{Z}_{o,i}^{-}$, respectively. The modified loss is then given by

$$\mathcal{L}_{\mathcal{G}} = \sum_{\boldsymbol{x}_i \in \mathbb{X}} \sum_{\boldsymbol{z}_\ell \in \mathbb{Z}_{o,i}} \left(C y_{i\ell}\, ln\left(\sigma(\boldsymbol{x}_i^\top \boldsymbol{u}_\ell)\right) + (1 - y_{i\ell}) \ln\left(1 - \sigma(\boldsymbol{x}_i^\top \boldsymbol{u}_\ell)\right)\right), \quad (3)$$

where $\mathbb{Z}_{o,i} = \mathbb{Z}_{o,i}^{+} \cup \mathbb{Z}_{o,i}^{-}$ includes both positive items and mined negative items associated with query $\boldsymbol{x}_i$.

Inference: An Approximate Nearest Neighbor Search (ANNS) index $\mathcal{A}$ is constructed over the item representations to facilitate zero-shot inference. This index encompasses either solely novel item representations for zero-shot inference or a combination of observed and novel items for generalized zero-shot inference. Thus, $\mathcal{A}$ is built over $\{\boldsymbol{z}_\ell \mid \boldsymbol{z}_\ell \in \mathbb{Z}_T\}$, where $\mathbb{Z}_T$ represents the set of items at inference, either $\mathbb{Z}_n$ or $\mathbb{Z} \cup \mathbb{Z}_n$.

During inference, the embedding of a query T is calculated as $\boldsymbol{x}_T = \mathcal{E}(X_T) \in \mathbb{R}^d$ with X_T being the query text. To retrieve relevant items for $\boldsymbol{x}_T$, $\mathcal{A}$ is queried using $\boldsymbol{x}_T$. The computational complexity of IRENE's inference is $\Omega\left(E + d \log\left(|\mathbb{Z}| + |\mathbb{Z}_n|\right)\right)$, where E is the cost of encoder $\mathcal{E}$. IRENE is designed to seamlessly incorporate new items by calculating their representations and adding them to the ANNS index [40] as and when novel they arrive in the system. This feature is particularly advantageous for applications like sponsored search, where novel items are frequently introduced. In such scenarios, re-training the model with a large volume of items can be resource-intensive, making IRENE a viable and efficient solution.

5 THEORETICAL GUARANTEES

Prior to evaluating the experimental efficacy of IRENE, we establish theoretical guarantees for the zero-shot generalization of EMMETT, with a specific focus on the case of IRENE. To simplify our analysis, we represent data samples as query-item pairs, $\boldsymbol{s} = (\boldsymbol{x}, \boldsymbol{z})$. The dataset $\mathbb{S} = \{\boldsymbol{s}\}$ is formed through two potential approaches. The most straightforward method involves randomly selecting N queries and items from the respective distributions $\mathcal{X}$ and $\mathcal{Z}$, pairing them to create $\hat{\mathbb{S}} = \{\boldsymbol{s}_i = (\boldsymbol{x}_i, \boldsymbol{z}_i) \mid i = 1, 2, \ldots N\}$. Alternatively, considering all possible query-item pairings from $\mathbb{X}$ and $\mathbb{Z}$ (as defined in Section 3.1), results in $\mathbb{S} = \{(\boldsymbol{x}_i, \boldsymbol{z}_\ell) \mid i = 1, 2, \ldots N, \ell = 1, 2, \ldots L\}$, meaning, $\boldsymbol{s}$ is drawn from the Cartesian product space $\mathcal{X} \times \mathcal{Z}$. The dataset is indexed by $j = L(i-1) + \ell,\ j = 1, 2, \ldots M = NL$. In this context, the extreme (meta) classification problem becomes binary, with target values y in $\mathcal{Y}$, where $y_j = 1$ indicates a positive association between item $\boldsymbol{z}_\ell$ and the query $\boldsymbol{x}_i$. Within the XC setting, it is well known that negatively associated pairs are significantly more likely to occur than positive ones. We assume that any set $\mathbb{S}$ contains at most κ positively associated pairs. We also assume norm bounds on the encoder representations of the queries, and the learnt classifiers, *i.e.*, $\max_{\boldsymbol{x} \in \mathcal{E}(\mathcal{X})} \|\boldsymbol{x}\|_2 \leq B$ and $\max_{\boldsymbol{w} \in \mathbb{W}} \|\boldsymbol{w}\|_2 \leq W$, respectively.

In assessing the generalization performance of EMMETT, we recall the concepts of empirical and true risks. Given a function $f(\cdot)$, and a chosen loss function, the empirical risk over a dataset $\mathbb{S}$, and the true risk are defined as:

$$\hat{\mathcal{R}} = \frac{1}{M} \sum_{\substack{j=1 \\ \boldsymbol{s}_j \sim \mathbb{S}}}^{M} \text{loss}(f(\boldsymbol{s}_j), y_j) \text{ and } \mathcal{R} = \mathbb{E}_{\boldsymbol{s} \sim \mathcal{X} \times \mathcal{Z}} \left[\text{loss}(f(\boldsymbol{s}), y)\right],$$

respectively. For our analysis, we use the weighted binary cross-entropy loss, $Cy \ln(f) + (1-y)\ln(1-f)$. We assume that the function f belongs to the class $\mathcal{F}$.

The generalizability of a model is indicated by the difference between empirical and true risks. This deviation depends on both data and the model. The model's influence is typically quantified using the *Rademacher complexity* $\mathfrak{R}$ [32] of the function class $\mathcal{F}$. This complexity can be evaluated empirically over the dataset (known as the empirical Rademacher complexity $\hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F})$), or across all datasets of size M, given by $\mathfrak{R}_M(\mathcal{F}) = \mathbb{E}_{\mathbb{S}}\left[\hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F})\right]$. Data dependence is often highlighted by bounding the risk deviation using the McDiarmid inequality [32]. The following theorem bounds the deviation of the true risk from the empirical risk.

Theorem 1. **(Generalization performance of EMMETT)** *Let R and $\hat{R}$ denote the true and empirical risk, respectively, and $\hat{\mathfrak{R}}_{\mathbb{S}}$ denote the empirical Rademacher complexity over set $\mathbb{S}$. Let $p \ll 1$ be the probability that a query-item pair is positively associated (i.e., $y_j = 1$), and q denote the probability that a set $\mathbb{S}$, $|\mathbb{S}| = M$ has at most κ positive pairs. Then, with probability at least $1-\delta$, $\delta \in (2q, 1)$, we have the following generalization bound:*

$$R \leq \hat{R} + \hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F}) + 3\left(q + \sqrt{\frac{\ln\left(\frac{2}{\delta - 2q}\right)}{2M}}\right), \text{ where} \tag{4}$$

$$q \leq \exp\left\{-2M\left(1 - p - \frac{\kappa}{M}\right)^2\right\}. \tag{5}$$

Proof. The detailed proof is provided in Appendix A of the Supplementary Document. We summarize the proof here. Without loss of generality, we redefine the loss to have the target labels $y_j \in \{-1, 1\}$, giving rise to the following form of the loss:

$$g(\boldsymbol{s}, y) = \text{loss}\left(f(\boldsymbol{s}), y\right) = \left(\frac{1-y}{2}\right) f(\boldsymbol{s}) - C\left(\frac{1+y}{2}\right) f(\boldsymbol{s}).$$

We then define the function $\Phi(\mathbb{S})$ as follows

$$\Phi(\mathbb{S}) = \sup_{f \in \mathcal{F}} \left\{\mathbb{E}\left[g(\boldsymbol{s}, y)\right] - \hat{\mathbb{E}}\left[g(\boldsymbol{s}_j, y_j)\right]\right\}.$$

By means of an extended McDiarmid inequality (that accounts for the positive-negative sample imbalances) [32], we get

$$\Pr\left(\left|\Phi(\mathbb{S}) - \mathbb{E}_{\mathbb{S}}\left[\Phi(\mathbb{S})\right]\right| \geq \epsilon\right) \leq 2q + 2\exp\left\{-2M\left(\epsilon - q\right)^2\right\} = \delta$$

$$\Rightarrow \epsilon = q + \sqrt{\frac{\ln\left(\frac{2}{\delta - 2q}\right)}{2M}}.$$

Given the inequality $\mathbb{E}_{\mathbb{S}}\left[\Phi(\mathbb{S})\right] \leq 2\mathfrak{R}_M(\text{loss} \circ \mathcal{F}) = 2\hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F})$ [32], substituting for δ into the first equation, and simplifying, we get

$$R \leq \hat{R} + \hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F}) + 3q + 3\sqrt{\frac{\ln\left(\frac{2}{\delta - 2q}\right)}{2M}}.$$

To bound q, we bound the probability that the set $\mathbb{S}$ contains at least $M - \kappa$ positively associated pairs by means of the Hoeffding inequality, which completes the proof of Theorem 1. □

Theorem 1 sheds light on the generalizability of new items in XC meta classifiers. Consistent with standard findings, we note that generalization gap is inversely related to the dataset size, M. Furthermore, the choice of dataset $\mathbb{S}$ over $\tilde{\mathbb{S}}$ reveals distinct advantages when considering the probabilities p and q. Specifically, a 1-vs-all setting, wherein we consider the Cartesian product space of queries and items, will lead to better overall performance. Further, for sufficiently small κ and sufficiently large M, we have $q \approx \exp\{-2M\}$. We also observe a direct correlation between generalization, and the complexity of the function class $\mathcal{F}$. In scenarios where $\mathcal{F}$ corresponds to the class of meta-classifier generators, the associated Rademacher complexity is detailed in the following lemma:

Lemma 2. *(***Rademacher complexity of the IRENE generator***) Let $\mathcal{F}$ be the class of functions defined in the IRENE algorithm, comprising pre-determined encoder representations and classifiers, a given classifier selector that outputs K classifiers, and $\mathcal{G}$, the meta-classifier generator. Then, the Rademacher complexity of $\mathcal{F}$ can be bounded as follows:*

$$\hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F}) \leq O\left(B\|\mathbf{M}\|_2\sqrt{d\ln(K+1)}\right), \tag{6}$$

where $\boldsymbol{x} \in \mathbb{R}^d$ and $\mathbf{M} \in \mathbb{R}^{d \times 1}$ is the weight matrix associated with the linear layer.

Proof. The detailed proof is provided in Appendix A of the Supplementary Document and follows by repeated application of Talagrand's lemma [32]:

$$\begin{aligned}
\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}) &\leq B\,\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_1) \\
&\leq B\|\mathbf{M}\|_2\,\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_2) \\
&\leq B\|\mathbf{M}\|_2\sqrt{d\ln(K+1)}\,\hat{\mathfrak{R}}_{\mathcal{S}}(\mathcal{F}_3) \\
&\leq O\left(B\|\mathbf{M}\|_2\sqrt{d\ln(K+1)}\right),
\end{aligned}$$

where $\mathcal{F}_3 = C_{lf}\left(\mathcal{S}(\boldsymbol{z}_\ell)\right)$, $\mathcal{F}_2 = \text{SelfAttn.}(\mathcal{F}_3)$ and $\mathcal{F}_1 = \mathbf{M}\mathcal{F}_2 + \boldsymbol{b}$. □

Lemma 2 yields several critical insights. Firstly, for a given XC base (consisting of an encoder and classifiers), the complexity of the meta-classifier increases logarithmically with K. This creates a balancing act: using a single classifier may result in inadequate model capacity, but increasing K increases model complexity, potentially degrading performance, and slowing down training. Ablation experiments presented in Section 7 validate this claim. Secondly, the complexity of the meta-classifier generator is independent of L, indicating that the model effectively scales in generalizing to novel labels. This scalability is attributed to the classifiers being fixed during the generator's training. The subsequent corollary discusses the Rademacher complexity of the function class $\mathcal{F}$ when the XC classifiers and the generator $\mathcal{G}$ are updated concurrently.

Corollary 3. (**Rademacher complexity of the IRENE generator with trainable classifier**) *Let $\mathcal{F}$ be the class of functions defined in the IRENE algorithm as in Lemma 2. Let the classifier set $\mathbb{W}$ be trainable over the meta-classifier loss. Then, the Rademacher complexity of $\mathcal{F}$ can be bounded as follows:*

$$\hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F}) \leq O\left(B^2 W \sqrt{\frac{L}{M}} \|\mathbf{M}\|_2 \sqrt{d \ln(K+1)}\right). \tag{7}$$

This result is obtained by combining Lemma 2 with an extension of Theorem 3 present in Awasthi et al. [4]. Training classifiers via the meta-classifier loss incurs a complexity in the order of $\sqrt{L}$. For the sake of completeness, we present the extension of Theorem 3 present in Awasthi et al. [4], relevant to the XC setting, with the model trained on N data points $\tilde{\mathbb{X}}$, and the loss defined over classifiers $\mathbb{W}$.

The following Lemma bounds the Rademcacher complexity of the XC classifier class:

Lemma 4. (**Rademacher complexity of the XC classifiers**) *(extension of Awasthi et al. [4], Theorem 3) Let $\mathcal{F}$ be the class of linear classifiers defined over the seen-item set $\mathbb{Z}_s$ in the classical XC setting (cf. Section 3.1), i.e., $\mathcal{F} = \{\langle \boldsymbol{x}, \boldsymbol{w}_\ell \rangle \mid \ell = 1, 2, \ldots, L\}$. Then, the Rademacher complexity of $\mathcal{F}$ can be bounded as $\hat{\mathfrak{R}}_{\mathbb{S}}(\mathcal{F}) \leq \frac{LBW}{\sqrt{N}}$, where $|\mathbb{X}| = N$ is the Cardinality of the training set.*

The proof is provided in Appendix A of the Supplementary Document. As expected, the complexity of the XC classifier, particularly in terms of generalizing to unseen labels, increases linearly with the number of labels. This is because the classifier's learning depends on the query-item pairs available during training. Therefore, as novel items arrive, additional data is necessary for accurately learning the classifiers.

These findings underscore the effectiveness of the meta-classifier-based EMMETT framework in achieving zero-shot generalization. They also validate the design choices implemented in the IRENE generator and algorithm, which contribute to a model with favorable model complexity. We now proceed to provide experimental evidence to support the IRENE algorithm, complemented by ablation studies. These studies elucidate our design decisions, connecting them to the theoretical bounds established earlier.

6 EXPERIMENTAL RESULTS

Datasets: We validate the performance of IRENE across a diverse set of datasets spanning multiple applications, and label space sizes (cf. Table 1). For LF-AmazonTitles-1.3M and LF-Wikipedia-500K, we use the experimental setup reported in Bhatia et al. [7], while for LF-AOL-270K and LF-WikiHierarchy-550K, we refer to [9]. We create the zero-shot versions of these datasets by randomly partitioning the set of items into observed and novel using a 90-10 split ratio [39].

Baselines: We demonstrate IRENE's efficacy by comparing its performance when built atop leading dense retrieval encoders such as NGAME [13], ANCE [47], MACLR [48], and DPR [23]. We also compare against competitive zero-shot XC methods such as SemSup-XC [2], ZestXML [17].

IRENE's Training Procedure: Given a base encoder $\mathcal{E}$, the classifiers $\mathbb{W}$ are trained using a one-versus-all BCE loss formulation similar to Renée [20]. However, unlike Renée, where the encoder and classifiers are trained jointly, we train only a single output-side transformer layer jointly with the classifiers. Subsequently, IRENE's meta-classifier generator block ($\mathcal{G}$) is trained using the hyper-parameters $K = 3$ and $D = 1$ for results reported in Table 2. Ablations on the choice of K and D are provided in Section 7.

Table 1: A summary of the datasets, and their corresponding applications, used in evaluating IRENE.

Dataset	Application Type	Feature Type
LF-AOL-270K	Query Completion	Short-Text
LF-Wikipedia-500K	Category Annotation	Long-Text
LF-WikiHierarchy-550K	Taxonomy Completion	Short-Text
LF-AmazonTitles-1.3M	Product Recommendation	Short-Text
KeywordPrediction-10M	Sponsored Search	Short-Text

Additional details on dataset creation and statistics, and detailed training procedures are provided in Appendices B and C, respectively, of the Supplementary Document.

Evaluation Setting and Metrics: Following [2, 17] we consider two settings: (i): zero-shot retrieval on novel items and (ii): generalized zero-shot retrieval on the combined set of observed and novel items. We report performance on metrics such as Precision@k (P@k), and Recall@k (R@k) at different truncation levels k.

Results on Benchmark Datasets: Table 2 presents comparisons of the zero-shot and generalized zero-shot performance of IRENE against the baselines. When added atop different encoders, IRENE consistently improves the performance in both the zero-shot (+1.5 to +40% in P@1 and +1 to +45% in R@10) and generalized evaluation settings (+1 to +29% in P@1 and +1 to +15% in R@10). IRENE shows larger gains on datasets such as LF-AOL-270K and LF-WikiHierarchy-550K where queries and their associated items have higher lexical dissimilarity. While modelling such relations can be challenging for small encoders, they nevertheless function as effective neighbour selectors, thereby identifying relevant classifiers for combining via IRENE's meta classifier generator. This is particularly visible in the case of MACLR, where IRENE improves P@1 by nearlly 40% in zero-shot on LF-WikiHierarchy-550K. IRENE's consistent gains over a wide range of applications on different encoders show the value of the information introduced by classifiers and emphasize the versatility of our approach.

When compared to zero-shot XC methods such as SemSup-XC, IRENE attains higher P@1 in the zero-shot setting (+10.31%) while being approximately 350 times more efficient than SemSup-XC at inference (cf. Table 3). These gains across various baselines establish the modularity of IRENE which can be used as a plug-and-play module atop any dense retriever to obtain more accurate representation for novel items for diverse recommendation applications.

Case-study on Sponsored Search: In the application of sponsored search, accurately matching user queries to billions of advertiser bid keywords presents a formidable challenge, exacerbated by varying bid amounts based on relevance and semantic relationship of the match. We demonstrate IRENE's efficacy by conducting offline experiments and online A/B tests on live search engine traffic. IRENE trained on the KeywordPrediciton-10M dataset was used to obtain representations for 100M novel keywords which were found to be 4% more accurate than those obtained from leading dense retrievers

Table 2: Comparison of zero-shot and generalized zero-shot accuracies of IRENE when applied to various baseline encoder frameworks. On average, IRENE improves P@1 by 10.1% and R@10 by 11.9% in the zero-shot setting. In the generalized setting IRENE boosts P@1 by 15.5% and R@10 by 11.5%. The best-performing algorithm amongst each pair of base encoder and its IRENE variant are indicated in boldfaced.

Model	LF-AOL-270K-10				LF-WikiHierarchy-550K-10				LF-AmazonTitles-1.3M-10				LF-Wikipedia-500K-10			
	Zero-Shot		Generalized		Zero-Shot		Generalized		Zero-Shot		Generalized		Zero-Shot		Generalized	
	P@1	R@10	P@1	R@10	P@1	R@10	P@1	R@10	P@1	R@10	P@1	R@10	P@1	R@10	P@1	R@10
NGAME	30.90	54.20	20.16	38.27	46.01	58.66	66.19	27.08	30.42	36.44	45.14	30.25	**46.96**	65.27	**81.86**	**69.58**
NGAME+IRENE	**36.47**	**59.57**	**35.11**	**52.30**	**69.29**	**80.40**	**91.33**	**40.09**	**31.56**	**38.83**	**47.77**	**31.49**	44.91	**67.79**	78.99	69.27
ANCE	33.43	67.84	22.63	49.72	43.06	56.28	68.76	25.89	22.38	30.72	27.65	17.31	30.67	58.91	42.91	43.39
ANCE+IRENE	**36.84**	67.82	**30.84**	**51.75**	**66.54**	**82.10**	**90.72**	**39.74**	**22.75**	**32.72**	**36.78**	**22.34**	**41.59**	**71.59**	**71.39**	**63.46**
MACLR	11.31	18.24	9.26	7.52	30.37	35.47	59.44	14.31	**21.93**	28.59	27.50	15.99	39.56	68.53	46.59	46.62
MACLR+IRENE	**34.32**	**61.29**	**30.40**	**44.71**	**69.45**	**81.43**	**88.81**	**38.37**	21.56	**28.77**	**31.49**	**18.85**	**44.64**	**73.05**	**70.52**	**62.82**
DPR	30.38	53.82	19.71	37.99	44.84	59.29	65.19	26.73	**31.10**	**40.98**	38.18	26.93	**42.90**	**71.20**	51.54	61.84
DPR+IRENE	**36.80**	**60.22**	**35.07**	**52.57**	**69.65**	**80.01**	**89.52**	**39.84**	30.49	40.31	**43.08**	**29.25**	42.19	70.50	**70.39**	**66.71**
TF-IDF	13.74	28.05	6.61	9.90	22.79	21.44	64.50	10.88	8.12	24.15	16.33	20.40	11.53	23.89	15.07	14.79
Zest-XML	9.34	25.91	26.34	26.57	13.97	17.48	68.86	22.13	5.42	5.58	41.36	22.87	2.62	14.73	60.16	45.15
SemSup-XC	26.27	36.31	26.12	23.92	57.45	46.81	90.51	28.37	11.28	11.68	25.13	15.21	46.60	57.08	54.20	38.08
DEXA	21.68	41.85	25.09	46.76	54.83	66.89	76.18	36.17	28.83	35.19	48.19	30.89	42.76	67.37	67.98	65.86

Table 3: Comparison of inference time (in ms) on a single V100 GPU for the LF-AmazonTitles-1.3M-10 dataset. This includes (i) generating the representation of an unseen item (Representation Time/ Rep. Time), and (ii) retrieving relevant items for a given query (Retrieval Time). Compared to dual encoder methods like NGAME and DEXA, IRENE adds no latency overhead, with the representation time increasing by 0.4 milliseconds. However, IRENE is about 350 times faster than SemSup-XC, which aggregate token-level similarities between queries and documents to obtain the final scores.

Method	**Rep. Time** (ms) ↓ (per item)	**Retrieval Time** (ms) ↓ (per query)
NGAME	0.08	0.43
SemSup-XC	N/A	151.51
DEXA	0.48	0.43
NGAME + IRENE	0.54	0.43

in production, when evaluated in terms of the R@100 metric. In live deployment, IRENE was found to increase the click-through rate (ad clicks obtained per unit query) and decrease the quick-back rate (fraction of users who quickly closed the ad) by 4.2% and 0.9%, respectively. In IRENE a novel keyword can be encoded in under 1ms and the approach demonstrates a 13% increase in good keyword predictions, as ascertained by expert judges. To handle potential distribution shifts that could occur in such dynamic settings, we can continually grow the base extreme classifier set by adding new item classifiers into it, as and when they receive clicks. Please refer to Appendix E of the Supplementary Document for detailed studies on IRENE's application to sponsored search.

IRENE's Computational Cost: Table 3 presents comparisons on the time taken for generating representations for a novel item and retrieving the relevant items given a query. When incorporating a novel item, NGAME performs a forward pass over a 6-layer DistilBERT. DEXA incurs an additional cost of an ANNS search over the cluster centroids. IRENE, on the other hand, incurs additional costs from the classifier selector $\mathcal{S}$ and meta-classifier generator $\mathcal{G}$, which involves an ANNS search over the set of observed items and a forward pass through a one-layer encoder respectively. The representation generation step was executed on a single NVIDIA Tesla V100 GPU for each method, while the ANNS search was carried out on a 96-core CPU machine.

While being efficient at inference, IRENE's classifier selector ($\mathcal{S}$) and meta-classifier generator ($\mathcal{G}$) incur minimal training costs. On a single NVIDIA Tesla V100 GPU, the training time for the NGAME encoder on the LF-AmazonTitles-1.3M-10 dataset was 83 hours and training IRENE on top of the NGAME encoder took only 6 hours.

7 ABLATIONS

We perform ablation experiments to evaluate the impact of various design choices within the components of the meta-classifier generator ($\mathcal{G}$) and the classifier selector ($\mathcal{S}$) in IRENE.

Meta-classifier Generator $\mathcal{G}$: Table 4 shows the effect of increasing the number of layers in IRENE's meta-classifier generator block ($\mathcal{G}$). Increasing the number of layers from 1 to 2 improves zero-shot P@1 by about 1%. However, in increasing $\mathcal{G}$'s depth to 4, the performance plateaus, which could be attributed to overfitting associated with increased complexity of $\mathcal{G}$. Additionally, we try out simpler alternatives, namely sum and weighted sum. IRENE performs better than these simpler alternatives by 24% P@1 in zero-shot evaluation underscoring the significance of IRENE's attention-based meta-classifier generator.

Classifier Selector $\mathcal{S}$: We we evaluate the effect of changing K, the number of observed items retrieved by $\mathcal{S}$, given an ANNS-based $\mathcal{S}$, on zero-shot performance. From the results presented in Table 4, we observe that increasing K beyond 3 causes the performance to initially plateau. Subsequently, increasing K to 20 brings about a decrease in performance. This is consistent with observations made in Section 5, wherein smaller K yield a tighter generalization bound (and therefore, superior performance), as derived in Lemma 2.

Inputs to the Meta-classifier Generator $\mathcal{G}$: Keeping the classifier selector $\mathcal{S}$ and meta-classifier generator $\mathcal{G}$ architecture fixed, the inputs passed to $\mathcal{G}$ were changed to encoder representations z of the selected observed items, instead of their classifiers w_ℓ. IRENE with classifiers w_ℓ of the selected observed items yields superior

Table 4: Ablation study on meta-classifier generator $\mathcal{G}$ and classifier selector $\mathcal{S}$ component in NGAME + IRENE on LF-WikiHierarchy-550K-10 dataset for zero-shot evaluation. The depth D denotes the number of layers in the transformer-based meta-classifier generator $\mathcal{S}$, while K denotes the numbers of neighbors selected by $\mathcal{S}$. We observe that that smaller values for $K \in \{2, 3, 6\}$ and $D \in \{1, 2\}$ yield superior results. We observe that setting $D = 1$ and $K = 3$ works reasonably well, balancing performance and the computational overhead in learning complex meta-classifier generators.

	Ablations	P@1 ↑	P@5 ↑	R@10 ↑
	IRENE ($D = 1, K = 3$)	69.29	38.81	80.40
Generator ($\mathcal{G}$)	$D = 2,\ K = 3$	70.36	39.06	80.39
	$D = 4,\ K = 3$	70.71	39.11	80.27
	$\mathcal{G}$ as Sum, $K = 3$	45.49	25.46	59.01
	$\mathcal{G}$ as wt. Sum, $K = 3$	46.39	25.88	59.29
Selector ($\mathcal{S}$)	$D = 1,\ K = 1$	68.98	38.56	80.11
	$D = 1,\ K = 2$	69.34	38.74	80.25
	$D = 1,\ K = 6$	69.99	39.12	80.79
	$D = 1,\ K = 20$	69.07	38.57	79.80

Table 5: Performance comparison of NGAME and NGAME+IRENE as the percentage of novel items (Novel Ratio) in the WikiHierarchy dataset is varied from 10% to 40%. While NGAME exhibits a significant decrease, NGAME+IRENE is relatively more resilient.

Novel Ratio	NGAME			NGAME+IRENE		
	P@1 ↑	P@5 ↑	R@10 ↑	P@1 ↑	P@5 ↑	R@10 ↑
10%	66.19	59.25	27.08	91.33	86.67	40.09
20%	65.29	58.22	26.66	89.37	85.29	39.33
30%	63.59	57.12	26.09	89.50	84.64	38.48
40%	60.12	53.47	24.45	87.64	82.59	36.99

performance, with a significant margin of 4% on P@1. This underscores the value of leveraging high-capacity classifiers of the selected items, which contain beneficial information for representing a novel item.

Detailed discussions regarding these ablations are present in Appendix D on the Supplementary Document.

Ratio of Novel Items: As we vary the percentage of novel items in the LF-WikiHierarchy-550K dataset from 10% to 40%, We observe that NGAME+IRENE is relatively more resilient to changes in the ratio of novel items, in comparison to the base NGAME algorithm. These results are summarized in Table 5.

Adapting to Few-Shot Scenarios - IRENE-OneShot: A plethora of algorithms have been proposed to address few-shot scenarios, with a particular emphasis on mitigating catastrophic learning. We study the extreme case of this where only a single ground truth query is revealed for an item. IRENE can be seamlessly extended to utilize the additional data available for such one-shot items by enhancing its classifier selector $\mathcal{S}$. IRENE leverages the revealed query of the one-shot item to fetch the nearest classifiers alongside the item itself. A max-voting strategy is then employed to improve the selection of observed classifiers over scenarios where only the item text is considered for this purpose. This variant is denoted as IRENE-OneShot. Remarkably, IRENE-OneShot can exploit the revealed data to improve the prediction accuracy without having the need to fine-tune the base model. To provide a comprehensive defense against catastrophic forgetting, we explore a theoretical baseline where NGAME is trained on a consolidated dataset comprising the initial training data for observed items and the data made available for the one-shot items. Even when compared to this ideal and optimal baseline, IRENE-OneShot was found to be superior in terms of P@1 by about 1%. Further, compared to Semsup-XC, which finetunes its encoder on the revealed data, IRENE-OneShot was at least 10% better in R@10. These results highlight the effectiveness of IRENE-OneShot, showcasing its capability to surpass models that fine-tune their encoders with revealed data, all the while preserving the base encoder, and reducing latency and complexity (cf. Table 6).

Table 6: Performance evaluation of (NGAME+) IRENE-OneShot, an extension for one-shot retrieval. One randomly selected query is made available to all algorithms for the novel items, and evaluation is done solely on the one-shot items. NGAME-retrained involves training NGAME method from scratch on the original training corpus along with the revealed queries for previously unseen items. Even with a significant training overhead for the NGAME-retrained model (~82 hours on LF-AmazonTitles-1.3M-10 and ~43 hours on LF-Wikipedia-500K-10), IRENE consistently outperforms the NGAME-retrained model by 1-2% across all metrics without incurring any additional training overhead. This is achieved by incorporating the new queries into the classifier selector module (at inference time) to refine the classifier shortlist.

Method	LF-AmazonTitles-1.3M-10			LF-Wikipedia-500K-10		
	P@1	P@5	R@10	P@1	P@5	R@10
NGAME-retrained	30.55	15.39	36.48	47.37	15.33	66.37
SemSup-XC-OneShot	13.96	6.98	19.76	46.25	13.77	57.04
IRENE-OneShot	**31.92**	**16.67**	**39.72**	**47.90**	**15.79**	**68.69**

8 CONCLUSIONS

In this paper, we studied the problem of large-scale zero-shot retrieval and developed techniques to efficiently and accurately represent novel items. We proposed a new algorithmic framework, EMMETT, for learning accurate meta-classifiers for novel items. EMMETT is a generic framework wherein different architectural choices and the training strategies can potentially yield algorithms with varying accuracy and efficiency trade-offs, presenting ample opportunities for future research explorations. We also developed a new algorithm, IRENE, that is simple, practically deployable and can significantly boost the performance of any Siamese encoder with minimal overheads. Finally, we developed a novel theoretical framework for analyzing the generalization performance in large-scale zero-short retrieval. Comprehensive empirical validation and online A/B tests in *Sponsored Search* application on a major search engine demonstrated the utility of IRENE and EMMETT.

REFERENCES

[1] G. Aggarwal, J. Feldman, and S. Muthukrishnan. 2006. Bidding to the top: VCG and equilibria of position-based auctions. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*.

[2] P. Aggarwal, A. Deshpande, and K. Narasimhan. 2023. SemSup-XC: Semantic Supervision for Zero and Few-shot Extreme Classification. In *ICML*.

[3] R. Agrawal, A. Gupta, Y. Prabhu, and M. Varma. 2013. Multi-label learning with millions of labels: Recommending advertiser bid phrases for web pages. In *WWW*.

[4] P. Awasthi, N. Frank, and M. Mohri. 2020. Adversarial Learning Guarantees for Linear Hypotheses and Neural Networks. In *Proceedings of the 37th International Conference on Machine Learning*, Vol. 119. 431–441. https://proceedings.mlr.press/v119/awasthi20a.html

[5] R. Babbar and B. Schölkopf. 2017. DiSMEC: Distributed Sparse Machines for Extreme Multi-label Classification. In *WSDM*.

[6] P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, and T. Wang. 2018. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. arXiv:1611.09268 [cs.CL]

[7] K. Bhatia, K. Dahiya, H. Jain, A. Mittal, Y. Prabhu, and M. Varma. 2016. The extreme classification repository: Multi-label datasets and code. http://manikvarma.org/downloads/XC/XMLRepository.html

[8] A. Z. Broder, P. Ciccolo, M. Fontoura, E. Gabrilovich, V. Josifovski, and L. Riedel. 2008. Search Advertising Using Web Relevance Feedback. In *CIKM*.

[9] A. Buvanesh, R. Chand, J. Prakash, B. Paliwal, M. Dhawan, N. Madan, D. Hada, V. Jain, S. Mehta, Y. Prabhu, M. Gupta, R. Ramjee, and M. Varma. 2024. Enhancing Tail Performance in Extreme Classifiers by Label Variance Reduction. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=6ARlSgun7J

[10] W.-C. Chang, D. Jiang, H.-F. Yu, C. H. Teo, J. Zhang, K. Zhong, K. Kolluri, Q. Hu, N. Shandilya, V. Ievgrafov, J. Singh, and I. S. Dhillon. 2021. Extreme Multi-label Learning for Semantic Matching in Product Search. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2643–2651.

[11] R. Combes. 2023. An Extension of McDiarmid's Inequality. arXiv:1511.05240 [cs.LG]

[12] K. Dahiya, A. Agarwal, D. Saini, K. Gururaj, J. Jiao, A. Singh, S. Agarwal, P. Kar, and M. Varma. 2021. SiameseXML: Siamese Networks meet Extreme Classifiers with 100M Labels. In *ICML*.

[13] K. Dahiya, N. Gupta, D. Saini, A. Soni, Y. Wang, K. Dave, J. Jiao, K. Gururaj, P. Dey, A. Singh, D. Hada, V. Jain, B. Paliwal, A. Mittal, S. Mehta, R. Ramjee, S. Agarwal, P. Kar, and M. Varma. 2023. NGAME: Negative Mining-aware Mini-batching for Extreme Classification. In *WSDM*.

[14] K. Dahiya, D. Saini, A. Mittal, A. Shaw, K. Dave, A. Soni, H. Jain, S. Agarwal, and M. Varma. 2021. DeepXML: A Deep Extreme Multi-Label Learning Framework Applied to Short Text Documents. In *WSDM*.

[15] K. Dahiya, S. Yadav, S. Sondhi, D. Saini, S. Mehta, J. Jiao, S. Agarwal, P. Kar, and M. Varma. 2023. Deep encoders with auxiliary parameters for extreme classification. In *KDD*.

[16] L. Gao, X. Ma, J. Lin, and J. Callan. 2023. Precise Zero-Shot Dense Retrieval without Relevance Labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1762–1777.

[17] N. Gupta, S. Bohra, Y. Prabhu, S. Purohit, and M. Varma. 2021. Generalized Zero-Shot Extreme Multi-label Learning. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[18] N. Gupta, P. H. Chen, H.-F. Yu, Cho-J. Hsieh, and I. S. Dhillon. 2022. ELIAS: End-to-End Learning to Index and Search in Large Output Spaces. In *NeurIPS*.

[19] H. Jain, V. Balasubramanian, B. Chunduri, and M. Varma. 2019. Slice: Scalable Linear Extreme Classifiers trained on 100 Million Labels for Related Searches. In *WSDM*.

[20] V. Jain, J. Prakash, D. Saini, J. Jiao, R. Ramjee, and M. Varma. 2023. Renée: End-to-end training of extreme classification models. *Proceedings of Machine Learning and Systems* (2023).

[21] T. Jiang, D. Wang, L. Sun, H. Yang, Z. Zhao, and F. Zhuang. 2021. LightXML: Transformer with Dynamic Negative Sampling for High-Performance Extreme Multi-label Text Classification. In *AAAI*.

[22] K. S. Jones. 2021. A statistical interpretation of term specificity and its application in retrieval. *J. Documentation* 60 (2021), 493–502. https://api.semanticscholar.org/CorpusID:2996187

[23] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-T. Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *EMNLP*.

[24] S. Khandagale, H. Xiao, and R. Babbar. 2020. Bonsai: diverse and shallow trees for extreme multi-label classification. *ML* (2020).

[25] S. Kharbanda, A. Banerjee, E. Schultheis, and R. Babbar. 2022. CascadeXML: Rethinking Transformers for End-to-end Multi-resolution Training in Extreme Multi-label Classification. In *NeurIPS*.

[26] O. Khattab and M. Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *SIGIR*.

[27] H. Kim, G. Papamakarios, and A. Mnih. 2021. The Lipschitz Constant of Self-Attention. In *Proceedings of the 38th International Conference on Machine Learning*, Vol. 139. 5562–5571. https://proceedings.mlr.press/v139/kim21i.html

[28] Y. Liu, X. Gao, and L. Gao, Q. Han. J. Shao. 2020. Label-activating framework for zero-shot learning. In *Neural Networks*, Vol. 121. 1–9.

[29] T. K. R. Medini, Q. Huang, Y. Wang, V. Mohan, and A. Shrivastava. 2019. Extreme Classification in Log Memory using Count-Min Sketch: A Case Study of Amazon Search with 50M Products. In *NeurIPS*.

[30] T. Mensink, E. Gavves, and C. G. M. Snoek. 2014. COSTA: Co-Occurrence Statistics for Zero-Shot Classification. In *CVPR*.

[31] A. Mittal, N. Sachdeva, S. Agrawal, S. Agarwal, P. Kar, and M. Varma. 2021. ECLARE: Extreme Classification with Label Graph Correlations. In *WWW*.

[32] M. Mohri, A. Rostamizadeh, and A. Talwalkar. 2012. *Foundations of Machine Learning*. MIT Press.

[33] Y. Qu, Y. Ding, J. Liu, K. Liu, R. Ren, W. X. Zhao, D. Dong, H. Wu, and H. Wang. 2021. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering.

[34] S. Robertson and H. Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrievals* 3, 4 (April 2009), 333–389. https://doi.org/10.1561/1500000019

[35] B. Romera-Paredes and P. H. S. Torr. 2015. An Embarrassingly Simple Approach to Zero-shot Learning. In *ICML*.

[36] P. Rusmevichientong, D. P. Williamson, and D. B. Shmoys. 2006. An optimization framework for finding revenue maximizing bid prices in keyword auctions. In *WWW*.

[37] D. Saini, A.K. Jain, K. Dave, J. Jiao, A. Singh, R. Zhang, and M. Varma. 2021. GalaXC: Graph Neural Networks with Labelwise Attention for Extreme Classification. In *WWW*.

[38] T. Shen, G. Long, X. Geng, C. Tao, T. Zhou, and D. Jiang. 2023. Large Language Models are Strong Zero-Shot Retriever. arXiv:2304.14233

[39] D. Simig, F. Petroni, P. Yanki, K. Popat, C. Du, S. Riedel, and M. Yazdani. 2022. Open Vocabulary Extreme Classification Using Generative Models. In *Findings of the Association for Computational Linguistics: ACL 2022*. 1561–1583. https://aclanthology.org/2022.findings-acl.123

[40] Aditi Singh, Suhas Jayaram Subramanya, Ravishankar Krishnaswamy, and Harsha Vardhan Simhadri. 2021. FreshDiskANN: A Fast and Accurate Graph-Based ANN Index for Streaming Similarity Search. arXiv:2105.09613 [cs.IR]

[41] J. J. Subramanya, F. Devvrit, H. V. Simhadri, R. Krishnawamy, and R. Kadekodi. 2019. DiskANN: Fast accurate billion-point nearest neighbor search on a single node. *Advances in Neural Information Processing Systems* 32 (2019).

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, Vol. 30. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[43] J. Vuckovic, A. Baratin, and R. Tachet des Combes. 2020. A Mathematical Theory of Attention. arXiv:2007.02876 [stat.ML]

[44] L. Wang, N. Yang, and F. Wei. 2023. Query2doc: Query Expansion with Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). 9414–9423. https://doi.org/10.18653/v1/2023.emnlp-main.585

[45] Y. Wang, J. Liu, Y. Wang, C. Tai, J. Shao, J. Ma, and C. Zhai. 2015. A noise-filtered under-sampling scheme for imbalanced classification. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*.

[46] J. Xin, C. Xiong, A. Srinivasan, A. Sharma, D. Jose, and P. Bennett. 2022. Zero-Shot Dense Retrieval with Momentum Adversarial Domain Invariant Representations. In *Findings of the Association for Computational Linguistics: ACL 2022*. 4008–4020.

[47] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. Bennett, J. Ahmed, and A. Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*.

[48] Y. Xiong, W.-C. Chang, C.-J. Hsieh, H.-F. Yu, and I. Dhillon. 2021. Extreme Zero-Shot Learning for Extreme Text Classification. arXiv:2112.08652 [cs.LG]

[49] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. [n. d.]. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[50] C. Yun, S. Bhojanapalli, A. Singh Rawat, S. Reddi, and S. Kumar. 2020. Are Transformers universal approximators of sequence-to-sequence functions?. In *International Conference on Learning Representations*. https://openreview.net/forum?id=ByxRM0Ntvr

[51] J. Zhang, W. C. Chang, H. F. Yu, and I. Dhillon. 2021. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. In *NeurIPS*.

[52] W. X. Zhao, J. Liu, R. Ren, and J.-R. Wen. 2023. Dense Text Retrieval based on Pretrained Language Models: A Survey. *ACM Trans. Inf. Syst.* (2023).