

RAGSys: Item-Cold-Start Recommender as RAG System

Emile Contal^{1,*†}, Garrin McGoldrick^{1,†}

¹Crossing Minds Inc, San Francisco, USA

Abstract

Large Language Models (LLM) hold immense promise for real-world applications, but their generic knowledge often falls short of domain-specific needs. Fine-tuning, a common approach, can suffer from catastrophic forgetting and hinder generalizability. In-Context Learning (ICL) offers an alternative, which can leverage Retrieval-Augmented Generation (RAG) to provide LLMs with relevant demonstrations for few-shot learning tasks. This paper explores the desired qualities of a demonstration retrieval system for ICL. We argue that ICL retrieval in this context resembles item-cold-start recommender systems, prioritizing discovery and maximizing information gain over strict relevance. We propose a novel evaluation method that measures the LLM's subsequent performance on NLP tasks, eliminating the need for subjective diversity scores. Our findings demonstrate the critical role of diversity and quality bias in retrieved demonstrations for effective ICL, and highlight the potential of recommender system techniques in this domain.

Keywords

Recommender systems, Information Retrieval, Large Language Models, Few-Shot Learning, In-Context-Learning

1. Introduction

Large Language Models (LLMs) have emerged as a powerful tool for natural language processing, demonstrating remarkable abilities in areas like text completion, summarization, and question answering [1]. One of their most intriguing capabilities is their potential to learn "common sense" – general knowledge about the world that allows them to reason and make inferences beyond the literal meaning of text. This has fueled excitement about the possibility of achieving zero-shot learning, where LLMs can solve unseen problems without any prior training on specific tasks [2].

However, a crucial distinction exists between generic public knowledge and the specific private knowledge required for most real-world use cases. While LLMs excel at generic text completion or chat-like interactions, practical applications often demand solving specific and repeatable downstream tasks within a particular domain [3]. This typically necessitates knowledge specific to a business or organization, such as understanding internal processes, up-to-date product details, or customer behavior.

Fine-tuning, a technique where LLMs are trained on large datasets tailored to the target task, offers a path towards adapting LLMs to these domain-specific needs. Yet, fine-tuning presents significant challenges. When trained on tasks-specific data, LLMs tend to forget knowledge and skills gained in the initial training, a phenomenon referred to as Catastrophic Forgetting [4]. Consequently, a fine-tuned LLM loses some of its ability to generalize to novel examples that aren't well represented in its fine-tuning training data. Moreover, while fine-tuning allows an LLM to memorize task-specific information, it doesn't necessarily allow the LLM to reason about that information [5]. As a final consideration, keeping LLMs constantly up-to-date using fine-tuning can be infeasible, especially for domains with frequently changing information like e-commerce product inventory, whereas it is easy to update a database in real-time from which information is retrieved.

As an alternative to fine-tuning, In-Context Learning (ICL) offers a promising approach for leveraging LLMs in

scenarios with limited data. This approach exploits the demonstrated ability of LLMs for "meta-learning" – essentially, learning how to learn. In [6], the authors prove the capacity of LLMs to effectively ingest in-context training data points and solve statistical optimization problems such as gradient descent. ICL enables practitioners to leverage Retrieval Augmented Generation (RAG), that is enriching the input prompt by information that is retrieved in real-time [7]. We refer to [8] for a recent survey on ICL.

This paper focuses on few-shot learning and the retrieval of relevant demonstrations for this process, where a demonstration is some text which is included in the LLM's context to demonstrate how the LLM should formulate correct answers. Few-shot learning presents a well-structured problem, allowing us to evaluate the quality of the retrieval algorithm using established classification metrics. Crucially, we show that enriching a language model with a few-shot example retriever offers a powerful method to achieve fine-tuning-like behavior, steering the output of the LLM towards the desired outcome even with limited data. Interestingly, increasing the context size in prompts beyond a certain point yields diminishing returns. The most impactful information resides within a relatively small set of well-chosen demonstrations, rather than overloading the prompt with vast amounts of data [9]. This highlights the importance of effective retrieval strategies, transforming k -shot learning into a top- k information retrieval problem at its core.

Building upon this concept, this paper identifies desirable properties for a RAG system under the framework of few-shot learning. We demonstrate that state-of-the-art retrieval systems in this context resemble item-cold-start recommender systems. Unlike exact search algorithms that prioritize precision and recall, our focus is on discovery, by maximizing the set of collective information gain from the retrieved demonstrations. This necessitates solving various trade-offs between query relevance, quality scoring, as well as diversity algorithms to ensure a variety of informative examples are surfaced. Furthermore, we propose a method for evaluating RAG system performance through the subsequent performance of the enriched LLM on established NLP tasks like question answering or text generation. This methodology offers a valuable approach to directly assessing diversity and quality-based retrieval systems, which removes the need to define a subjective diversity score, a historically challenging aspect of evaluating such systems in academic settings [10].

Information Retrieval's Role in RAG Systems (IR-RAG) - 2024

*Corresponding author.

†Both authors contributed equally to this research.

✉ emile@crossingminds.com (E. Contal);

garrin.mcgoldrick@crossingminds.com (G. McGoldrick)



© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

To summarize, in this paper we study the impact of diversity and quality bias in retrieving demonstrations for ICL. We start by reviewing the use of diversity and other biases in both ICL and Information Retrieval works. We then propose a method for evaluating the performance of different retrieval algorithms. Then we present experiments and results demonstrating the impact of diversity and quality bias on an LLM’s ability to generate correct answers. Finally we discuss the applicability of state-of-the-art ICL retrieval algorithms in real-world setting, and show that recommendation engines offer a better solution than semantic search engines.

2. Related Work

This paper sits at the intersection of two distinct, but increasingly intertwined, research areas: In-Context Learning for Large Language Models and Information Retrieval. While ICL focuses on enabling LLMs to learn from carefully selected contextual information, IR deals with retrieving relevant information from document collections. Our work leverages concepts from both fields to address the challenge of few-shot learning with LLMs.

2.1. Few-Shot In-Context Learning with Retrieval Augmented Generation

Within the context of RAG, few-shot learning can be defined as a specific scenario where the “documents” retrieved are actually “examples” used to guide the LLM. These examples can also be referred to interchangeably as “In-Context Examples” (ICE) or “demonstrations”. The importance of ICL in achieving state-of-the-art LLM performance is undeniable, with its ubiquitous presence in top benchmarks across various domains. Consequently, ICL research is a rapidly evolving field with numerous proposed algorithms.

2.1.1. Pure Diversity

Several noteworthy ICL approaches have emerged that address the challenge of retrieving informative examples for few-shot learning. Some methods only promote the diversity of the demonstrations, like in [11] where the authors utilize k -means clustering in a dense embeddings space to achieve diversity. By applying k -means to the sentence embeddings of the demonstrations, this approach ensures that the retrieved examples cover a variety of semantic spaces, inherently increasing the mutual information of the retrieved set, but without taking into account the relevancy to the query.

2.1.2. Pure Quality

Other approaches focus on identifying examples where the LLM exhibits low token-level uncertainty. In [12] the authors analyze token probabilities within candidate 0-shot prompts. By prioritizing prompts where the LLM has the highest generation likelihood (low perplexity), this approach aims to select examples that hold the potential for significant learning gains for the LLM. The intuition that the authors give is that a prompt that is more expected by the LLM is more likely to help it extracting the relevant information. Accessing the per-token probabilities for all examples incurs a significant compute cost, but can be pre-computed as they do not depend on the query.

2.1.3. Pure Relevance

Notably, a connection can be drawn between traditional full-text search algorithms and pure relevance approaches. In [13] the authors use BM25 [14], a well-established retrieval ranking function commonly used in information retrieval tasks. This approach essentially leverages the strengths of BM25 in identifying examples with terms highly similar to the query, to select the most relevant examples for the specific task at hand. This strategy ensures the retrieved examples are topically relevant to the task while potentially introducing some variation in the specific phrasing or wording used.

Finally, neural ranking, one of the most commonly used ICL approach, typically yielding superior results [15, 16, 17, 18, 19], is maximizing similarity in a dense embeddings space. These methods, like KATE [20], utilize k -Nearest Neighbors (kNN) search using the cosine distance of sentence embeddings to retrieve the examples most semantically similar to the prompt. Scaling this method leverages vector search algorithms, commonly used in large-scale information retrieval tasks, where efficient retrieval of semantically similar documents is crucial.

While general purpose pre-trained embeddings like BERT [21] form a strong baseline, learning specific embeddings for retrieval, and in particular ICL, is a very active area of research. In [22] the authors build upon BERT and introduce CoBERT that improves the retrieval performances by an order of magnitude. Other embeddings model have been proposed for ICL retrieval in [16] and [17]. Some authors also explored training full language models [15], as well as LLM [19], showing further improvements compared to traditional embedding-based approaches. While leading to superior results, these supervised neural ranking models for learning-to-rank necessitate orders of magnitude more training examples data, that is typically not available to practitioners. In addition, without any explicit metric space such as dense embeddings, efficient retrieval indexing such as [23] cannot be used.

2.1.4. Diversity/Relevance Trade-off

While both relevance and diversity are crucial for effective few-shot learning, methods yielding the best ICL results combine these two paradigms rather than prioritizing one over the other. This is achieved by maximizing a carefully balanced trade-off between semantic similarity to the prompt and diversity of the retrieved examples. Unsupervised techniques can be adapted to prioritize the selection of examples that are both relevant to the prompt and dissimilar to each other. In [24] the authors introduce a greedy method to select relevant demonstrations while ensuring enough coverage. They define specific coverage strategies adapted to the problem of program generation. In [25] the authors employ an active learning setting, where a voting algorithm selects a set of examples penalizing the top- k closest from already selected ones, using cosine distance in an embedding space.

The most popular unsupervised approach for achieving this balance between relevance and diversity is Maximal Marginal Relevance (MMR). MMR retrieves a set of examples by iteratively selecting the example that maximizes a linear combination of the scores of relevance to the prompt and dissimilarity to the previously retrieved examples. It was analyzed in [26] for ICL and was shown to outperform

simpler methods. Alternatively to MMR, Determinantal Point Processes (DPP) has been used in [18] to optimize the joint information of the selected k examples. However, exactly solving the DPP optimization being NP-hard, hence the authors also employ greedy maximization.

2.2. Diversity in Information Retrieval and Online Learning

The concept of diversity in information retrieval has been a long-running topic of research. In this section we propose a short review of the use of diversity in the IR literature and related domains.

2.2.1. In Information Retrieval

The MMR algorithm was analyzed in [27], and compared against other approaches like KL-divergence optimization in [28]. Pure retrieval algorithms typically optimize for recall, not information maximization. Agreeing on a diversity objective function remains a challenge. Diversity is sometimes introduced as a heuristic to cover possible interpretations of the same query, instead of minimizing information overlap from near-duplicated results. In [10] the authors leverage a concept of *information nuggets* with documents to estimate the redundancy of the set of retrieved documents. Topic modeling is also employed, such as [29] that uses a taxonomy of categories labelling the documents and the queries. The desired properties of diverse retrieval are furthermore characterized in [30]. A various set of similarity methods and diversification algorithms are analyzed in [31] on sparse features vectors. Among diversity evaluation methods based on topic modelling, three notable criteria used in the TREC Diversity track [32], ERR-IA [33], α -nDCG@ k [10], and NRRP [34], are compared in [35].

2.2.2. In Recommender Systems

Within IR, the recommender system literature brings an additional point-of-view on studying diversity in retrieval, by focusing on the benefit of diverse results for a user, instead of evaluating methods against a potentially arbitrary relevance/diversity trade-off. The difficulty of evaluating the impact of diversity, and the necessity for large scale real-world recommendation studies has been explored in [36]. In [37] and [38] the authors model the user behavior conditioned on the set of retrieved items. In [39] the authors improve the diversity versus relevance trade-off in recommender systems by directly learning a ranking model that favors diversity, instead of only applying diversity re-ranking methods.

2.2.3. In Online Learning

Learning a trade-off between relevancy and diversity also naturally occurs in related online frameworks such as active learning, multi-armed bandits and Bayesian optimization. In [40] the authors modify a learning-to-rank algorithm from users feedback, to inherently learn diverse rankings and demonstrate a positive impact on the original relevance metric. Other approaches such as [41] also introduce diversity in learning-to-rank algorithms while preserving the offline settings, but then are limited to evaluate using direct diversity measures.

Within batch-mode Bayesian optimization, in [42] and [43] the authors analyze two greedy exploration/exploitation algorithms to select the next batch of items maximizing the cumulative reward. Like with recommender systems, these online settings exemplify the theoretical and empirical importance of diversifying the selected set of items despite the true objective function only including pure relevance, the cumulative reward.

2.3. Quality Biases in Information Retrieval

Complementing the discussion on diversity in information retrieval, quality bias also plays a crucial role in effective retrieval. Quality bias refers to the prioritization of documents or examples considered to be more reliable or informative within the retrieved set. Incorporating quality considerations into retrieval algorithms can significantly improve standard unbiased IR metrics.

Several approaches have been explored to address quality bias in pure IR tasks. These can be broadly categorized into content-based and graph-based methods.

2.3.1. Content-Based Quality Biases

Content-based methods leverage existing signals inside the documents themselves to identify potentially lower-quality content. Examples include spam detection scores developed in works like [44] and [45]. By incorporating such scores during retrieval, the system can prioritize higher quality documents. More sophisticated content-based approaches don't limit at spam classification, but extract more generic quality features the content of documents. Works like [46] explore features such as stop-word statistics or entropy of the documents to generate quality scores. The authors demonstrate that biasing standard retrieval using these features leads to improved retrieval effectiveness even using unbiased IR metrics like nDCG.

2.3.2. Graph-Based Quality Biases

Instead of relying on the content itself, graph-based algorithms inherently capture implicit quality signals within their ranking model. PageRank, a seminal algorithm for web search ranking introduced in [47], exemplifies this approach. PageRank leverages the links structure between web articles to assign higher importance to web pages that are linked to by other high-quality pages. This process implicitly prioritizes documents with a higher perceived quality based on the quality of their in-links.

2.3.3. Connections to Recommender Systems

Interestingly, the concept of inherent quality bias in graph-based IR approaches resembles collaborative filtering techniques employed in recommender systems. In an analogous manner to learning-to-rank on a (item, item) graph, collaborative filtering addresses learning-to-rank on a bipartite (user, item) graph. In this way, collaborative filtering also implicitly learns a trade-off between item similarity and popularity, favoring items that are both similar to the user's past preferences and also generally well-received by other users.

3. Methodology

We propose to frame the ICL problem as an item-cold-start recommendation problem, where the query is an unseen item, and the objective is to retrieve from the pool of candidate few-shot demonstrations a set of items maximizing the cumulative reward to the user (the LLM). In this case, the reward is a measure of how much the retrieved items increase the probability that the LLM generates a correct answer. A solution to this optimization problem requires not only relevance, but also diversity and quality in the retrieved items, such that the amount of useful information presented to the LLM in the context is maximized.

Further, we propose to measure the impact of diversity on the retrieved items by directly calculating the probability of the LLM generating a correct answer given the context items. This is in contrast to a typical retrieval context where the retriever is evaluated by calculating some metric relating to the accuracy and recall of documents most similar to the query. In such a setting, it is typical to add a term to the metric which measures the diversity of the retrieved documents to promote more diverse retrievers, knowing that diversity improves the reward to the user but without having an explicit model connecting diversity to the user’s reward. In the case of retrieving demonstrations for inclusion in an LLM’s context, we can directly measure the impact of diversity on the LLM’s reward by calculating the probability of the LLM generating a correct answer.

3.1. Problem Statement

Consider an answer a that should be generated by an LLM in response to a query q . The query can be a simple question such as "Who was the first man to walk on the moon?", or more general message such as "I'd like to find red shoes". The answer could take on many forms, such a factual response "Neil Armstrong", a clarifying question "What size and style of shoe are you looking for?", a JSON payload to send to an API `{"search_terms": ["red", "shoe"]}` etc.

Consider a set of demonstrations \mathcal{D} , where each demonstration is a pair (q, a) containing a query q and correct answer a , or a triple (q, a, \bar{a}) which additionally contains an incorrect answer \bar{a} . Datasets under this later triplet form are commonly used in Contrastive Learning approaches. We call C , a subset of demonstrations retrieved from \mathcal{D} , the context.

$$C \subset \mathcal{D} = \{(q_i, a_i, \bar{a}_i), \dots, (q_n, a_n, \bar{a}_n)\}$$

Given an auto-regressive LLM M , the query q , and a retrieved context C , we define $p_M(a | q, C)$ the probability that M generates the answer a . In practice, the tokens of the examples from the context C are appended to the tokens of the query q , using prompt formatting techniques that may be optimized to a specific LLM.

Putting it all together, for an unseen query q and unseen correct answer a , a few-shot retriever $R_{\mathcal{D}}$ must efficiently retrieve a subset of k demonstrations $R_{\mathcal{D}}(q) \in \mathcal{D}^k$ such that $p_M(a | q, R_{\mathcal{D}}(q))$ is maximized.

3.2. Evaluation

Consider the probability of generating the correct answer a given an empty context $p_M(a | q)$. We are interested in evaluating how much the context C increases the probability

of the correct answer a . That is, we want a metric which is related to difference of $p_M(a | q, C)$ and $p_M(a | q)$.

In a pure retrieval setting, we would be interested in finding the context C which contains the k demonstrations that are most similar to the q . And we could argue that if there exists a smooth function $f : q \rightarrow a$ which maps a query to its correct answer, then by retrieving the demonstrations whose queries are nearest to q , we should also be retrieving the answers which are closest to a , and this should help the language model M generate the correct answer a .

However, it is doubtful that the space in which q is compared to the demonstrations is one in which the function $f : q \rightarrow a$ is smooth, so it is not necessarily true that the retrieved answers are closest to a . Nor is it necessarily true that $p_M(a | q, C)$ is maximized when C contains those answers closest to a . Consider that the answer a might depend on some information which isn’t contained in a or any nearby answer.

Therefore, we prefer to measure $p_M(a | q, C)$ directly. In practice, given that M is an auto-regressive language model, this is done by taking the product of the probability of each token generated by M . The model generates text sequentially by predicting one token at a time based on the previously generated tokens. Let $a = (a_1, a_2, \dots, a_n)$ represent a sequence of tokens produced by the model. The probability of the model generating the sequence a can be expressed as the joint probability of generating each token in the sequence, conditioned on the tokens that precede it. This can be mathematically represented as:

$$p(a) = p(a_1) \cdot p(a_2|a_1) \cdot p(a_3|a_1, a_2) \cdots p(a_n|a_1, a_2, \dots, a_{n-1})$$

Thus, $p_M(a | q, C)$ is the product of the conditional probabilities of each token, and these probabilities are output by the LLM at inference time and are readily available in APIs serving LLMs such as the OpenAI API.

3.2.1. Classification Metrics

In binary classification, accuracy is typically used as an evaluation metric, and can be defined as:

$$\frac{1}{|\mathcal{D}|} \sum_{(x, y, \bar{y}) \in \mathcal{D}} \mathbf{1}(p(y | x) > p(\bar{y} | x))$$

Where: $|\mathcal{D}|$ is the number of examples in the dataset \mathcal{D} ; $\mathbf{1}(\cdot)$ is the indicator function that returns 1 if the condition is true and 0 otherwise; and $y(\bar{y})$ is the correct (incorrect) label for example x .

Given a retriever $R_{\mathcal{D}}$ and a demonstration $(q, a, \bar{a}) \in \mathcal{D}$, we introduce the simplified leave-one-out notation $R(q) = R_{\mathcal{D} \setminus \{(q, a, \bar{a})\}}(q)$. We define the metric MC1 which is related to accuracy:

$$\text{MC1} = \frac{1}{|\mathcal{D}|} \sum_{(q, a, \bar{a}) \in \mathcal{D}} \mathbf{1}(p_M(a | q, R(q)) > p_M(\bar{a} | q, R(q)))$$

In the case that many incorrect answers are provided for each query \bar{a} , we can extend this in the same manner as multi-class classification by requiring that the correct answer have greater probability than all the incorrect answers:

$$\frac{1}{|\mathcal{D}|} \sum_{(q, a, \bar{a}) \in \mathcal{D}} \prod_{\bar{a} \in \bar{a}} \mathbf{1}(p_M(a | q, R(q)) > p_M(\bar{a} | q, R(q)))$$

We also define a metric MC2 which extends this further to the case that multiple correct answers \mathbf{a} and multiple

incorrect answers \bar{a} are provided for each query. This metric is the average number of correct answers which have greater probability than all incorrect answers.

$$\text{MC2} = \frac{1}{|\mathcal{D}|} \sum_{(q, \mathbf{a}, \bar{\mathbf{a}}) \in \mathcal{D}} \frac{1}{|\mathbf{a}|} \sum_{a \in \mathbf{a}} \prod_{\bar{a} \in \bar{\mathbf{a}}} \mathbf{1}(p_M(a | q, R(q)) > p_M(\bar{a} | q, R(q)))$$

Finally, we define the related metric MC3. This metric is the ratio of probability of correct answers to the probability of incorrect answers.

$$\text{MC3} = \frac{1}{|\mathcal{D}|} \sum_{(q, \mathbf{a}, \bar{\mathbf{a}}) \in \mathcal{D}} \frac{\sum_{a \in \mathbf{a}} p_M(a | q, R(q))}{\sum_{\bar{a} \in \bar{\mathbf{a}}} p_M(\bar{a} | q, R(q))}$$

These metrics and their names follow those defined in [48]. While they are easy to interpret, these metrics are not well normalized: they don't take into account all possible correct and incorrect answers. As a result, if the sample of correct and incorrect answers have varying lengths and use of rare vocabulary tokens, these will impact the metrics.

3.2.2. Direct Preference Optimization Metric

We postulate that an ideal metric should obey the following properties: it should be positive when the retrieved context increases the probability of a correct answer; it should be equal in magnitude when the probability of a correct answer halves or doubles; it should relate to the probability of getting all correct answers such that if any one correct answer is impossible, the metric is minimized. Moreover, in the case that incorrect answers are provided, it should be positive when the context C increases the probability of correct answers more than that of incorrect answers.

We define the DPO metric as the negative of the Direct Preference Optimization loss [49], which satisfies these properties:

$$\text{DPO} = \log \sigma \left(\log \frac{p_M(a | q, R(q))}{p_M(a | q)} - \log \frac{p_M(\bar{a} | q, R(q))}{p_M(\bar{a} | q)} \right)$$

In the case that incorrect answers are not available, the term containing \bar{a} can be omitted while preserving the aforementioned properties.

Because the metric is proportional to probability ratio $p_M(a | q, C) / p_M(a | q)$ rather than the absolute probability $p_M(a | q, C)$, it is invariant to the number of tokens and frequency of rare vocabulary tokens in the answer. If this were not the case, then the score for an example would get a positive (negative) bias if the correct (incorrect) answer is shorter. Similarly, the score across a set of examples would weigh examples with shorter answers more strongly.

Another aspect of the DPO metric that is worth considering is that by using this metric to optimize the retriever, we are effectively fine-tuning a model. Consider the LLM model $p_M(a | q)$ which assigns a probability of a generation a given a prompt q . Now consider another model $p'_M(a | q')$, where $q' = R(q)$. From this perspective, we can consider that p'_M is functionally the same as p_M but with added parameters arising from R . And so, by finding a retriever R which maximizes the DPO metric, we are in effect fine-tuning the model p'_M .

3.3. Retrieval Algorithm

This section details the core algorithm employed for few-shot demonstration retrieval, which leverages a greedy strategy to maximize a combination of three key scores: query relevance, demonstration diversity, and demonstration quality bias. The full retrieval algorithm is presented in Algorithm 1.

3.3.1. Query Relevance

The relevance score between the query and each candidate demonstration is calculated using the cosine similarity of their respective BERT embeddings [21]. By computing the cosine similarity between the query embedding and the embedding of each demonstration's query, we obtain a score that reflects the topical similarity and semantic alignment between the query and the candidate demonstration.

3.3.2. Retrieved Demonstrations Diversity

To promote diversity in the retrieved demonstrations and avoid redundancy, we incorporate the Maximal Marginal Relevance (MMR) algorithm. MMR iteratively selects the items that maximizes the combined score of relevance to the query and dissimilarity to the previously chosen items. This ensures a balance between retrieving relevant items and ensuring they cover a variety of information. A parameter, λ_d , is used to control the trade-off between relevance and diversity. Higher values of λ_d prioritize relevance, whereas lower values prioritize diversity.

3.3.3. Demonstration Quality Bias

While the pre-trained BERT embeddings capture semantic relationships, they do not inherently account for the quality of the few-shot demonstrations. To address this, we explicitly introduce a demonstration quality bias term related to the popularity of an item in a training dataset. This score is computed using the log perplexity of the demonstration answer a , given the demonstration question q .

$$\frac{1}{|a|} \sum_{a_i \in a} \log p_M(a_i | q)$$

This can be interpreted as measuring the probability of the correct answer a given the query q , normalized to the length of the answer. This can also be interpreted as a proxy for a popularity bias, akin to the number of connections of an item in graph-based retrieval algorithms like recommender systems. Like in the article [12], the intuition is that the more frequently a related sequence of tokens occurs in the pre-training dataset of the LLM, the more likely the model will be able to extract its relevant information. Rather than directly analyzing the massive amount of text data (often trillions of tokens) used to pre-train the LLM, we focus on the perplexity of the sequence. Perplexity acts as a proxy, indicating how surprised the LLM is by the sequence, essentially, how well it aligns with what the LLM expects to see. A parameter λ_b controls the trade-off between relevance/diversity and quality bias. Lower values of λ_b emphasize high-quality demonstrations.

Algorithm 1 MMR with quality bias

Require: $1 \leq k \leq n$; $0 \leq \lambda_d \leq 1$; $0 \leq \lambda_b \leq 1$
Require: $Q \in \mathbb{R}^d$ \triangleright query embedding
Require: $\forall 1 \leq i \leq n, E_i \in \mathbb{R}^d$ \triangleright example question embedding
Require: $\forall 1 \leq i \leq n, b_i \in \mathbb{R}$ \triangleright example quality bias
 $Q \leftarrow \frac{Q}{\|Q\|}$; $\forall 1 \leq i \leq n, E_i \leftarrow \frac{E_i}{\|E_i\|}$
 $\forall 1 \leq i \leq n, v_i \leftarrow \lambda_b Q E_i^\top + (1 - \lambda_b) b_i$
 $C_1 \leftarrow \arg \max_i v_i$
for $2 \leq s \leq k$ **do**
 $\forall 1 \leq i \leq n, m_i \leftarrow \max_{1 \leq j < s} E_i E_{C_j}^\top$
 $\forall 1 \leq i \leq n, w_i \leftarrow \lambda_d v_i - (1 - \lambda_d) m_i$
 $C_s \leftarrow \arg \max_{i \notin \{C_1, \dots, C_{s-1}\}} w_i$
end for

4. Experiments

4.1. Experimental Setup

4.1.1. The Dataset Choice

We are interested in a publicly available dataset which meets the following criteria: it should have enough statistical power so that we can resolve small differences in accuracy, ideally it will have hundreds of examples or more; it doesn't need to have vast amounts of data as this isn't a typical setting for few-shot learning, and the cost of conducting experiments can become burdensome; it should provide correct and incorrect answers so that we can report classification metrics from Section 3.2.1; it should be big enough to contain similar examples with partially redundant information so the use of diversity can improve collective information presented to the LLM in a context.

4.1.2. The TruthfulQA Dataset

We chose to conduct our experiments using the TruthfulQA dataset [48] which meets these requirements. The dataset contains 817 distinct examples, which yields a standard error in the range of 1% to 2% for accuracy measures in the range of 90% to 50%. Each example contains a single query, and a variable number of correct and incorrect answers. And by considering each distinct (q, a) pair as a demonstration for the purpose of building a context, the retriever is faced with similar demonstrations as multiple (q, a) pairs share the same query (on average, the dataset contains 3.5 correct answers for each query).

4.1.3. Generating Demonstrations Pairs and Triplets

The dataset is used in three different ways in this paper:

- \mathcal{D}_{MC} : this is the dataset as described in [48]. It contains 817 examples, each of which contains a variable number of correct and incorrect answers. The metrics MC1, MC2 and MC3, which can accept an arbitrary number of correct and incorrect answers as inputs, are calculated over this dataset.
- \mathcal{D}_{DPO} : this is the set of every distinct (q, a, \bar{a}) triple contained in \mathcal{D}_{MC} . It contains 12,485 such triplets. The DPO metric is calculated over this dataset.
- \mathcal{D}_{ICL} : this is the set of every distinct (q, a) pairs contained in \mathcal{D}_{MC} . It contains 2,846 such pairs. This is the set of demonstrations from which a context is drawn. That is, $C \subset \mathcal{D}_{ICL}$.

When calculating a metric score for an example $(q_i, \mathbf{a}_i, \bar{\mathbf{a}}_i)$, all demonstrations with the query q_i are left out from the demonstrations available for inclusion in the context. In this manner, the correct answers \mathbf{a}_i are not included in the context when the LLM is presented with query q_i .

4.1.4. The Language Models

We conducted our experiments using four noteworthy LLMs: the smaller base text-completion model *Mistral-7B-v0.1* (7B parameters), and the larger instruct-fine-tuned mixture of models *Mixtral-8x22B-Instruct-v0.1* (141B parameters) from Mistral [50]; as well as a smaller chat-tuned model *Llama-3-8B-chat* (8B parameters) and a larger chat-tuned model *Llama-3-70B-chat* (70B parameters) from Llama ¹.

All four models are open-weights LLMs, meaning their internal parameters are publicly available for scrutiny and potential fine-tuning. These modern models stand out for achieving impressive performance on various tasks despite their relatively compact size. This efficiency makes it an attractive option for resource-constrained environments where deploying colossal models might not be feasible.

4.2. Implementation Details and Ablation Study

We implemented the Algorithm 1 and metrics from Section 3.2.1 in python. We computed BERT embeddings using the package *sentence_transformers* ², and implemented the retrieval algorithms in *numpy*. We queried all LLM models using the Together API ³.

We did not perform hyper-parameter tuning, and fixed the two parameters to $\lambda_d = 0.75$ and $\lambda_b = 0.95$ in all experiments. We fixed the amount of retrieved demonstrations to $k = 6$, matching the number of few-shot examples from the fixed primer example from the TruthfulQA paper [48].

To measure the impact of the separated components of Algorithm 1, relevance, diversity, and bias, we implemented variants of the retrieval algorithm using only one or two of the three components:

- Fix: fixed primer examples [48]
- Bias: Pure quality bias [12]
- Rel: Pure semantic similarity [20] (KATE)
- Rel+Bias: Semantic similarity plus quality bias
- Rel+Div: Semantic similarity plus diversity [26]
- Rel+Bias+Div: Algorithm 1

4.3. Main Results

We present the experimental metrics for the 6 retrievers for the 4 different LLMs: in Table 1 and 2 for the Mistral models, and Table 3 and 4 for the Llama-3 models.

Our evaluation relies on a combination of metrics to assess the effectiveness of different retrieval strategies for ICL. The normalized DPO metric provides the most valuable insights for each LLM individually but cannot be directly compared across models. The three additional classification metrics allow for objective performance comparisons across models. However, these metrics are susceptible to bias based on token sequence length.

¹<https://llama.meta.com/llama3/>

²<https://sbert.net/>

³<https://docs.together.ai/docs/inference-python>

Table 1
Evaluation Metrics with *Mistral-7B-v0.1*

Method	DPO	MC1	MC2	MC3
Fix	-20.40	0.2815	0.2086	0.4285
Bias	-33.56	0.2411	0.1652	0.3596
Rel	-12.71	0.4455	0.3664	0.5925
Rel+Bias	-13.63	0.4602	0.3663	0.5969
Rel+Div	-12.37	0.5177	0.3930	0.6616
Rel+Div+Bias	-14.54	0.4676	0.3592	0.6255

Table 3
Evaluation Metrics with *Llama-3-8B-chat*

Method	DPO	MC1	MC2	MC3
Fix	-22.12	0.3623	0.2709	0.5195
Bias	-17.55	0.3831	0.2876	0.5729
Rel	-17.20	0.4920	0.4046	0.6518
Rel+Bias	-17.14	0.5043	0.4083	0.6570
Rel+Div	-16.14	0.5520	0.4173	0.7009
Rel+Div+Bias	-15.80	0.5177	0.4007	0.6841

The impact of few-shot learning is best seen by comparing the three MC metrics for *Rel+Div* for a smaller model against *Fix* for a larger model: the smaller models (7B and 8B parameters) enriched with ICL RAG are essentially matching or outperforming the bigger models (141B and 70B parameters) without ICL RAG.

The results consistently demonstrate that incorporating both relevance and diversity into the retrieval strategy leads to superior performance across all metrics and for both LLMs. For all models, and for all metrics, *Rel+Div* largely outperforms *Rel*. This finding reinforces the importance of not just retrieving relevant demonstrations but also ensuring a diverse set that maximizes the informative value for the LLM.

Interestingly, the impact of the low perplexity bias yields contrasting results. For both Mistral models, adding this bias results in a decline in performance on almost all metrics. Conversely, both Llama-3 models exhibit overall improvement with the low perplexity bias, in particular with the DPO metric. This intriguing observation suggests that LLM-dependent hyper-parameter tuning of λ_b might be necessary to optimize retrieval strategies for specific models. Alternatively, the low perplexity bias itself may benefit from further refinement. Using an opposite intuition, we may argue that instead of prioritizing demonstrations the LLM already finds likely, introducing demonstrations that surprise the model the most could be beneficial for certain LLMs, potentially maximizing the learning impact of each demonstration. These findings open exciting new avenues for future research in ICL retrieval strategies, creating a parallel with novelty and serendipity concepts in recommender systems.

4.4. Calibrating Diversity using DPO

Calibrating the amount of diversity in the retrieved set is crucial when optimizing ICL retrieval. We highlight the difficulty of achieving this without our proposed methodology by demonstrating the non-monotonous relationship between the amount of diversity in the retrieved demonstrations and the resulting benefit to the LLM performance. To quantify diversity, we calculate the average cosine simi-

Table 2
Evaluation Metrics with *Mixtral-8x22B-Instruct-v0.1*

Method	DPO	MC1	MC2	MC3
Fix	-19.06	0.5202	0.3896	0.6799
Bias	-27.30	0.4382	0.3096	0.5948
Rel	-15.17	0.6193	0.5004	0.7616
Rel+Bias	-14.77	0.6389	0.5080	0.7657
Rel+Div	-12.67	0.6879	0.5181	0.8092
Rel+Div+Bias	-13.29	0.6573	0.5071	0.7924

Table 4
Evaluation Metrics with *Llama-3-70B-chat*

Method	DPO	MC1	MC2	MC3
Fix	-23.05	0.4382	0.3375	0.6184
Bias	-20.41	0.4455	0.3303	0.6424
Rel	-19.02	0.5483	0.4482	0.6958
Rel+Bias	-19.09	0.5532	0.4495	0.7054
Rel+Div	-13.93	0.6389	0.4834	0.7758
Rel+Div+Bias	-13.72	0.6022	0.4621	0.7583

larity between the BERT embeddings of each demonstration pair within the retrieved set. The LLM’s benefit is measured using the DPO metric. We then systematically vary λ_d while keeping the LLM fixed (*Llama-3-8B-chat*), the quality bias fixed ($\lambda_b = 0.95$), and the number of retrieved demonstrations constant ($k = 6$) to observe the empirical correlation between diversity and DPO. The results are visualized in Figure 1. This experiment underscores the importance of a metric measuring the impact of the retrieved context on the LLM, like DPO. Without such a metric, it would be challenging to effectively calibrate λ_d and achieve the optimal balance between relevance and diversity in the retrieved demonstrations.

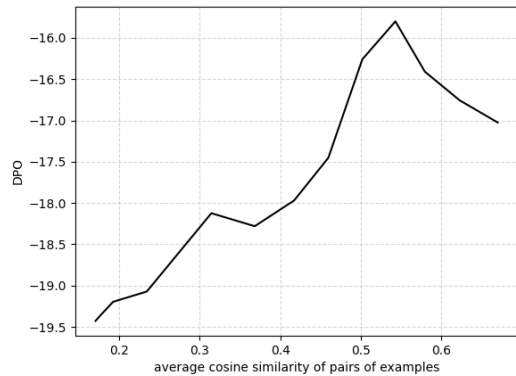


Figure 1: Diversity Metric and DPO. Non-monotonous relationship between a diversity metric, the average cosine similarity between embedding pairs, and the quality metric DPO. Obtained by varying λ_d with *Llama-3-8B-chat* and $k = 6$.

5. Discussion: Real-World RAG Systems

While the importance of diversity in ICL retrieval is paramount, we note that readily available RAG systems

rarely implement it directly within the core retrieval algorithm. There are several practical considerations to keep in mind for successful deployment.

5.1. Balancing Performance and Efficiency

Retrieval latency is crucial at scale. Exhaustive, brute-force nearest neighbor search is computationally expensive and impractical. Instead, real-world systems leverage efficient indexing techniques and approximated kNN algorithms, as described in [23], to ensure fast retrieval times. This approach is essential for handling large datasets while maintaining responsiveness. To seamlessly integrate with existing retrieval engines and leverage their optimized search capabilities, a retrieval algorithm for RAG must ensure its data is stored in a format compatible with these engines. Commonly indexed data structures include text itself or low-dimensional dense vector embeddings. By adhering to these indexing practices, RAG systems can effectively leverage the power of existing retrieval engines and achieve fast, scalable retrieval of informative examples.

5.2. ICL RAG versus Fine-Tuning

The computational cost of ICL may be evaluated against the cost of fine-tuning. For instance, consider a large LLM like gpt-3.5 with a current price 6x larger per input tokens between fine-tuned or default model⁴. While ICL requires additional input tokens, it is guaranteed to offer cost savings compared to fine-tuning when $k < 6$ with this model.

An interesting contrast between ICL and fine-tuning is highlighted in the paper [51]. The paper argues that fine-tuning can be more efficient than few-shot ICL in terms of cost and latency due to the super-linear increase in LLM latency with growing prompt sizes. However, this latency concern is less relevant with inference-throughput optimized LLM systems built with large GPU clusters, such as commonly used APIs. In these systems, the observed latency remains independent of the prompt size. From a latency perspective, adding ICL demonstrations can be considered free. Additionally, the paper suggests that ICL requires scanning through 100% of the demonstrations at query time. However this does not hold when employing real retrieval engines with indexing and approximate kNN, which significantly reduce the number of examples scanned during retrieval.

Furthermore, building a curated database of few-shot demonstrations offers significant advantages to practitioners. These demonstrations are not specific to a single LLM but can be readily utilized with any LLM architecture. This eliminates vendor lock-in and lets practitioners leverage the best LLM for the task at hand without concerns about compatibility. Perhaps even more importantly, a well-maintained database of few-shot examples automatically benefits from the continuous advancements in LLM technology. As newer, more powerful pre-trained LLMs become rapidly available, existing demonstrations can be used to enrich them quickly. This ensures applications leverage the latest capabilities without the need to completely re-engineer workflows. This reusability and adaptability position our few-shot learning engine as a powerful tool for harnessing the ever-evolving potential of LLMs to solve real business challenges.

⁴In May 2024, the price of gpt-turbo-0125 is \$0.5/M input tokens, \$1.5/M output tokens; and fine-tuned price of \$3/M input tokens, \$6/M output tokens, \$8M fine-tuning tokens

5.3. Achieving State-of-the-Art Retrieval with Available Tools

Traditional full-text search algorithms like BM25 lead to empirically lower ICL quality. Vector stores offer a more suitable solution for efficient retrieval based on semantic similarity. Numerous vendors provide vector store solutions, and they can be broadly categorized as follows:

In-Memory vector indexes, such as FAISS and nmslib, offer exceptional speed with minimal setup complexity, but limited scalability for larger datasets. They may not implement in-place addition or deletion of the indexed vectors. Self-Hosted vector databases, such as Elasticsearch and Postgres, provide a balance between scalability and performance, at a much larger setup complexity. They typically implement efficient addition and deletion of the indexed vectors. SaaS vector stores, such as Pinecone and VertexAI, offer a convenient option with pre-configured infrastructure and almost no setup complexity. We invite the reader to consult the lists of integrated vector stores of LangChain⁵ and LlamaIndex⁶ for near-exhaustive lists of available tools.

Due to the complexities of incorporating such rules directly within retrieval indexing algorithm [52], none of the solutions known to the authors from any of the above category implements diversity or quality biasing of the result. A common heuristic to mitigate this problem is to retrieve a larger set of candidate examples (e.g., double the desired number) and then apply diversity techniques like MMR on the retrieved candidates as a post-processing step. Quality biasing can be indirectly achieved by modifying the indexed embeddings themselves. For instance, reducing the norm of embeddings associated with low-quality content can nudge the retrieval algorithm towards higher-quality examples. An exact implementation in the context of cosine-similarity or dot-product relevance is to add an additional column storing the quality bias, and set the corresponding value to 1 in the embedding of the query.

While vector search offers a powerful foundation for practical ICL retrieval, it often lacks native support for essential considerations like diversity or quality bias. These aspects are crucial for ensuring informative and effective retrieval of few-shot learning examples. Existing tools for recommendation engines, on the other hand, often excel in these areas. Recommendation engines natively incorporate rules that promote diversity by recommending a variety of items, or quality bias by prioritizing most popular products. Future research directions as well as practical systems for ICL retrieval could explore adapting or integrating these well-established techniques from recommender systems to further enhance the effectiveness and sophistication of few-shot learning through information retrieval. State-of-the-art ICL for real-world applications can be achieved by combining the strengths of vector search with the established "diversity-aware" retrieval approaches from recommender systems.

6. Conclusion

This paper explored the critical role of information retrieval in ICL for few-shot learning with Large Language Models. Our work identified key desirable properties for ICL

⁵<https://python.langchain.com/docs/integrations/vectorstores/>

⁶https://docs.llamaindex.ai/en/stable/module_guides/storing/vector_stores/

retrieval systems. We demonstrated that state-of-the-art retrieval in this domain resembles recommender systems under the item cold-start problems. Unlike traditional information retrieval prioritizing for exact recall, our approach emphasizes discovery by maximizing the collective information gain from retrieved demonstrations. This necessitates balancing query relevance, quality scoring, and diversity to ensure a variety of informative examples are surfaced. Furthermore, we propose a novel evaluation method for ICL retrieval based on the subsequent performance of the enriched LLM on NLP tasks. This approach eliminates the need for subjective diversity scores, a challenge in information retrieval evaluation. Our findings demonstrate the significant impact of diversity and quality bias in retrieving demonstrations for ICL. By incorporating these well-established techniques from recommender systems, we can unlock the full potential of ICL for few-shot learning and empower LLMs to tackle real-world tasks with limited data.

7. Acknowledgments

To Ching-Wei Chen, for finding the name *RAGSys*, and the entire Crossing Minds team for their support during this research.

References

- [1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, *Advances in neural information processing systems* 33 (2020) 1877–1901.
- [2] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, Y. Iwasawa, Large language models are zero-shot reasoners, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), *Advances in Neural Information Processing Systems*, volume 35, Curran Associates, Inc., 2022, pp. 22199–22213. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/8bb0d291acd4ac60ef112099c16f326-Paper-Conference.pdf.
- [3] E. J. Hu, yelong shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-rank adaptation of large language models, in: *International Conference on Learning Representations*, 2022. URL: <https://openreview.net/forum?id=nZeVKeeFYf9>.
- [4] Y. Luo, Z. Yang, F. Meng, Y. Li, J. Zhou, Y. Zhang, An empirical study of catastrophic forgetting in large language models during continual fine-tuning, *ArXiv abs/2308.08747* (2023). URL: <https://api.semanticscholar.org/CorpusID:261031244>.
- [5] L. Berglund, M. Tong, M. Kaufmann, M. Balesni, A. C. Stickland, T. Korbak, O. Evans, The reversal curse: LLMs trained on "a is b" fail to learn "b is a", *ArXiv abs/2309.12288* (2023). URL: <https://api.semanticscholar.org/CorpusID:262083829>.
- [6] Y. Bai, F. Chen, H. Wang, C. Xiong, S. Mei, Transformers as statisticians: Provable in-context learning with in-context algorithm selection, in: A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), *Advances in Neural Information Processing Systems*, volume 36, Curran Associates, Inc., 2023, pp. 57125–57211. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/b2e63e36c57e153b9015fece2352a9f9-Paper-Conference.pdf.
- [7] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlga, A. Shashua, K. Leyton-Brown, Y. Shoham, In-Context Retrieval-Augmented Language Models, *Transactions of the Association for Computational Linguistics* 11 (2023) 1316–1331. doi:10.1162/tacl_a_00605.
- [8] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, L. Li, Z. Sui, A survey for in-context learning, *CoRR abs/2301.00234* (2023).
- [9] D. Machlab, R. Battle, LLM In-Context Recall is Prompt Dependent, *CoRR* (2024) arXiv:2404.08865. doi:10.48550/arXiv.2404.08865. arXiv:2404.08865.
- [10] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, I. MacKinnon, Novelty and diversity in information retrieval evaluation, in: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '08*, Association for Computing Machinery, New York, NY, USA, 2008, p. 659–666. doi:10.1145/1390334.1390446.
- [11] W. Yu, D. Iter, S. Wang, Y. Xu, M. Ju, S. Sanyal, C. Zhu, M. Zeng, M. Jiang, Generate rather than retrieve: Large language models are strong context generators, in: *International Conference for Learning Representation (ICLR)*, 2023.
- [12] H. Gonen, S. Iyer, T. Blevins, N. Smith, L. Zettlemoyer, Demystifying prompts in language models via perplexity estimation, in: H. Bouamor, J. Pino, K. Bali (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, Association for Computational Linguistics, Singapore, 2023, pp. 10136–10148. doi:10.18653/v1/2023.findings-emnlp.679.
- [13] S. Agrawal, C. Zhou, M. Lewis, L. Zettlemoyer, M. Ghazvininejad, In-context examples selection for machine translation, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 8857–8873. doi:10.18653/v1/2023.findings-acl.564.
- [14] S. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford, Okapi at trec-3, in: *Overview of the Third Text REtrieval Conference (TREC-3)*, Gaithersburg, MD: NIST, 1995, pp. 109–126. URL: <https://www.microsoft.com/en-us/research/publication/okapi-at-trec-3/>.
- [15] O. Rubin, J. Herzig, J. Berant, Learning to retrieve prompts for in-context learning, in: M. Carpuat, M.-C. de Marneffe, I. V. Meza Ruiz (Eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Seattle, United States, 2022, pp. 2655–2671. doi:10.18653/v1/2022.naacl-main.191.
- [16] L. Wang, N. Yang, X. Huang, B. Jiao, L. Yang, D. Jiang, R. Majumder, F. Wei, Text Embeddings by Weakly-Supervised Contrastive Pre-training, *CoRR* (2022) arXiv:2212.03533. doi:10.48550/arXiv.2212.03533. arXiv:2212.03533.
- [17] X. Li, K. Lv, H. Yan, T. Lin, W. Zhu, Y. Ni, G. Xie, X. Wang, X. Qiu, Unified demonstration retriever for in-context learning, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), *Proceedings of the 61st Annual*

- Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 4644–4668. doi:10.18653/v1/2023.acl-long.256.
- [18] J. Ye, Z. Wu, J. Feng, T. Yu, L. Kong, Compositional exemplars for in-context learning, in: A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, J. Scarlett (Eds.), Proceedings of the 40th International Conference on Machine Learning, volume 202 of *Proceedings of Machine Learning Research*, PMLR, 2023, pp. 39818–39833. URL: <https://proceedings.mlr.press/v202/ye23c.html>.
- [19] L. Wang, N. Yang, F. Wei, Learning to retrieve in-context examples for large language models, in: Y. Graham, M. Purver (Eds.), Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, St. Julian's, Malta, 2024, pp. 1752–1767.
- [20] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, W. Chen, What makes good in-context examples for GPT-3?, in: E. Agirre, M. Apidianaki, I. Vulić (Eds.), Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures, Association for Computational Linguistics, Dublin, Ireland and Online, 2022, pp. 100–114. doi:10.18653/v1/2022.deeLIO-1.10.
- [21] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.
- [22] O. Khattab, M. Zaharia, Colbert: Efficient and effective passage search via contextualized late interaction over bert, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 39–48. doi:10.1145/3397271.3401075.
- [23] A. Beygelzimer, S. Kakade, J. Langford, Cover trees for nearest neighbor, in: Proceedings of the 23rd International Conference on Machine Learning, ICML '06, Association for Computing Machinery, New York, NY, USA, 2006, p. 97–104. doi:10.1145/1143844.1143857.
- [24] I. Levy, B. Bogin, J. Berant, Diverse demonstrations improve in-context compositional generalization, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 1401–1422. doi:10.18653/v1/2023.acl-long.78.
- [25] H. Su, J. Kasai, C. H. Wu, W. Shi, T. Wang, J. Xin, R. Z. 0037, M. Ostendorf, L. Zettlemoyer, N. A. Smith, T. Y. 0009, Selective annotation makes language models better few-shot learners, in: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, OpenReview.net, 2023. URL: <https://openreview.net/pdf?id=qY1hlv7gwg>.
- [26] X. Ye, S. Iyer, A. Celikyilmaz, V. Stoyanov, G. Durrett, R. Pasunuru, Complementary explanations for effective in-context learning, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Findings of the Association for Computational Linguistics: ACL 2023, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 4469–4484. doi:10.18653/v1/2023.findings-acl.273.
- [27] J. Carbonell, J. Goldstein, The use of mmr, diversity-based reranking for reordering documents and producing summaries, in: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '98, Association for Computing Machinery, New York, NY, USA, 1998, p. 335–336. doi:10.1145/290941.291025.
- [28] C. X. Zhai, W. W. Cohen, J. Lafferty, Beyond independent relevance: methods and evaluation metrics for subtopic retrieval, in: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, SIGIR '03, Association for Computing Machinery, New York, NY, USA, 2003, p. 10–17. doi:10.1145/860435.860440.
- [29] R. Agrawal, S. Gollapudi, A. Halverson, S. Jeong, Diversifying search results, in: Proceedings of the Second ACM International Conference on Web Search and Data Mining, WSDM '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 5–14. URL: <https://doi.org/10.1145/1498759.1498766>. doi:10.1145/1498759.1498766.
- [30] S. Gollapudi, A. Sharma, An axiomatic approach for result diversification, in: Proceedings of the 18th International Conference on World Wide Web, WWW '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 381–390. doi:10.1145/1526709.1526761.
- [31] M. R. Vieira, H. L. Razente, M. C. N. Barioni, M. Hadjieleftheriou, D. Srivastava, C. Traina, V. J. Tsotras, On query result diversification, in: Proceedings of the 2011 IEEE 27th International Conference on Data Engineering, ICDE '11, IEEE Computer Society, USA, 2011, p. 1163–1174. doi:10.1109/ICDE.2011.5767846.
- [32] C. L. Clarke, N. Craswell, I. Soboroff, Overview of the trec 2009 web track., in: Trec, volume 9, 2009, pp. 20–29.
- [33] O. Chapelle, D. Metzler, Y. Zhang, P. Grinspan, Expected reciprocal rank for graded relevance, in: Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM '09, Association for Computing Machinery, New York, NY, USA, 2009, p. 621–630. doi:10.1145/1645953.1646033.
- [34] C. L. Clarke, M. Kolla, O. Vechtomova, An effectiveness measure for ambiguous and underspecified queries, in: Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory, ICTIR '09, Springer-Verlag, Berlin, Heidelberg, 2009, p. 188–199. doi:10.1007/978-3-642-04417-5_17.
- [35] C. L. Clarke, N. Craswell, I. Soboroff, A. Ashkan, A comparative analysis of cascade measures for novelty and diversity, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 75–84. doi:10.1145/1935826.1935847.
- [36] C.-N. Ziegler, S. M. McNee, J. A. Konstan, G. Lausen,

- Improving recommendation lists through topic diversification, in: Proceedings of the 14th International Conference on World Wide Web, WWW '05, Association for Computing Machinery, New York, NY, USA, 2005, p. 22–32. doi:10.1145/1060745.1060754.
- [37] S. Vargas, P. Castells, Rank and relevance in novelty and diversity metrics for recommender systems, in: Proceedings of the Fifth ACM Conference on Recommender Systems, RecSys '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 109–116. doi:10.1145/2043932.2043955.
- [38] S. Vargas, Novelty and diversity enhancement and evaluation in recommender systems and information retrieval, in: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 1281. doi:10.1145/2600428.2610382.
- [39] Y. Zheng, C. Gao, L. Chen, D. Jin, Y. Li, Dgcn: Diversified recommendation with graph convolutional networks, in: Proceedings of the Web Conference 2021, WWW '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 401–412. doi:10.1145/3442381.3449835.
- [40] F. Radlinski, R. Kleinberg, T. Joachims, Learning diverse rankings with multi-armed bandits, in: Proceedings of the 25th International Conference on Machine Learning, ICML '08, Association for Computing Machinery, New York, NY, USA, 2008, p. 784–791. doi:10.1145/1390156.1390255.
- [41] Y. Zhu, Y. Lan, J. Guo, X. Cheng, S. Niu, Learning for search result diversification, in: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 293–302. doi:10.1145/2600428.2609634.
- [42] T. Desautels, A. Krause, J. W. Burdick, Parallelizing exploration-exploitation tradeoffs in gaussian process bandit optimization, *Journal of Machine Learning Research* 15 (2014) 4053–4103. URL: <http://jmlr.org/papers/v15/desautels14a.html>.
- [43] E. Contal, D. Buffoni, A. Robicquet, N. Vayatis, Parallel gaussian process optimization with upper confidence bound and pure exploration, in: H. Blockeel, K. Kersting, S. Nijssen, F. Železný (Eds.), *Machine Learning and Knowledge Discovery in Databases*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 225–240.
- [44] A. Ntoulas, M. Najork, M. Manasse, D. Fetterly, Detecting spam web pages through content analysis, in: Proceedings of the 15th International Conference on World Wide Web, WWW '06, Association for Computing Machinery, New York, NY, USA, 2006, p. 83–92. doi:10.1145/1135777.1135794.
- [45] G. V. Cormack, M. D. Smucker, C. L. A. Clarke, Efficient and effective spam filtering and re-ranking for large web datasets, *Inf. Retr.* 14 (2011) 441–465. doi:10.1007/s10791-011-9162-z.
- [46] M. Bendersky, W. B. Croft, Y. Diao, Quality-biased ranking of web documents, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM '11, Association for Computing Machinery, New York, NY, USA, 2011, p. 95–104. doi:10.1145/1935826.1935849.
- [47] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, *Computer Networks and ISDN Systems* 30 (1998) 107–117. URL: <https://www.sciencedirect.com/science/article/pii/S016975529800110X>, proceedings of the Seventh International World Wide Web Conference.
- [48] S. C. Lin, J. Hilton, O. Evans, Truthfulqa: Measuring how models mimic human falsehoods, in: Annual Meeting of the Association for Computational Linguistics, 2021. URL: <https://api.semanticscholar.org/CorpusID:237532606>.
- [49] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, C. Finn, Direct preference optimization: Your language model is secretly a reward model, in: A. Oh, T. Neumann, A. Globerson, K. Saenko, M. Hardt, S. Levine (Eds.), *Advances in Neural Information Processing Systems*, volume 36, Curran Associates, Inc., 2023, pp. 53728–53741. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/a85b405ed65c6477a4fe8302b5e06ce7-Paper-Conference.pdf.
- [50] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. Singh Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. Renard Lavaud, M.-A. Lachaux, P. Stock, T. Le Scao, T. Lavril, T. Wang, T. Lacroix, W. El Sayed, Mistral 7B, arXiv e-prints (2023) arXiv:2310.06825. doi:10.48550/arXiv.2310.06825. arXiv:2310.06825.
- [51] H. Liu, D. Tam, M. Muqeeth, J. Mohta, T. Huang, M. Bansal, C. A. Raffel, Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), *Advances in Neural Information Processing Systems*, volume 35, Curran Associates, Inc., 2022, pp. 1950–1965. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/0cde695b83bd186c1fd456302888454c-Paper-Conference.pdf.
- [52] L. Li, C.-Y. Chan, Efficient indexing for diverse query results, *Proc. VLDB Endow.* 6 (2013) 745–756. doi:10.14778/2536360.2536373.