DRAG: Diversity in Retrieval Augmented Generation through the Application of Submodular Functions

Maria Paula Cortes-Lemos

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Science

University of Washington

2025

Committee:

Jeffrey Bilmes

Gina Anne Levow

Program Authorized to Offer Degree:

Department of Linguistics

University of Washington

Abstract

DRAG: Diversity in Retrieval Augmented Generation
through the Application of Submodular Functions

Maria Paula Cortes-Lemos

Chair of the Supervisory Committee:
Prof. Jeffrey Bilmes
Department of Electrical & Computer Engineering

This thesis applies a submodular approach to the reranking stage of Retrieval Augmented Generation to balance the relevance and diversity of the retrieved documents. After initial retrieval using *Contriever*, we experiment with submodular functions and a baseline of Maximal Marginal Relevance (MMR), a standard function for balancing relevance and diversity. We apply convex combinations of three approaches: 1) a submodular feature based function using LOG1P concavity and Facility Location, 2) One-Hot Quantization (a quantized modular function with a one-hot feature based function) with manual weights and Facility Location, and 3) One-Hot Quantization with exponential weight decay and Facility Location. We perform hyperparameter selection for the submodular functions and for MMR. We evaluate these on five datasets designed for diversity-focused tasks (news, politics, analogies, etc). We show submodular functions outperform or match MMR's performance in nearly all cases, with recall improvements exceeding 20% (relative difference) in the best case scenario. These results suggest a submodular approach can be effective to improve RAG systems, particularly in diversity-sensitive tasks.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I am immensely grateful to my advisor, Professor Jeffrey Bilmes, for his guidance and motivation throughout this work. His knowledge on submodular functions and encouragement to explore and question different approaches were invaluable in shaping this research. I thank the members of the Melodi Lab for creating a collaborative and fruitful research environment. Special recognition goes to Gantavya and Arnav for their technical discussions, developing a large fraction of the code for this project, and their friendship throughout this journey. I also thank Professor Gina Anne Levow for serving on my committee and providing thoughtful feedback. Finally, I am grateful to my family and friends for their support during my graduate studies.

DEDICATION

to mom, dad, Anto y Car. Y mi Chui y Bordito. Y a mis amigxs que están y estuvieron. Gracias por tanto.

Chapter 1

INTRODUCTION

Despite the rapid advancements in Large Language Models (LLMs) in the past few years, these models still have significant issues that can be detrimental to the quality and accuracy of their output. For instance, popular LLMs such as ChatGPT, Claude, Gemini, and others are prone to hallucinations, "a phenomenon in which the generated content appears nonsensical or unfaithful to the provided source content" [Huang et al., 2025], which cause the models to confidently output incorrect information. Similarly, vanilla LLMs (i.e., those without external augmentation or fine-tuning) are also susceptible to return answers based on outdated knowledge, as the information in the data they were trained and evaluated on becomes obsolete [Dhingra et al., 2022]. Furthermore, it is possible for LLMs to reflect societal biases from their training data and provide skewed information on controversial issues—which can cause serious ethical and practical challenges [Guo et al., 2024]. In a general sense, what all these limitations have in common is their relation to the data the LLM is built on and the way it is used by the model.

This is why *Retrieval Augmented Generation* (RAG), which is based on feeding information from an external database into the LLM, was conceived as a multi-purpose solution. Although RAG has shown promising results, there remain significant opportunities for improvement [Rau et al., 2024]. In their survey on RAG for large language models, Gao et al. [2024] mention low precision and recall in the retrieval step, outdated information in the context for the model, and redundancy and repetition when multiple retrieved passages have similar information. This last problem can be particularly detrimental if the goal is to make the system efficient and unbiased. It can lead to over-spent resources retrieving the same information over and over from different passages, as well as a lack of diversity in the retrieved

information, which can impact the quality of responses to queries that benefit from heterogeneous knowledge. Ideally, RAG systems should be able to contain as much non-redundant information as possible, as this would make the (limited) context richer.

This is why one of the proposed routes for the refinement of RAG is the consideration of diversity in the information retrieved from the external corpus. Introducing diversity can help improve the completeness and fairness of the answer [Rau et al., 2024], as well as help avoid redundancy in the retrieved context.

For this reason, there is growing interest in adapting RAG systems to account not only for similarity, but also for diversity. While this can be achieved at several stages throughout the RAG pipeline, this thesis focuses on diversity during retrieval. By modifying the function used to select context passages, we can guide retrieval toward more heterogeneous content. This approach creates a richer information subset while also avoiding the computational waste of retrieving and processing redundant passages with similar information, which ideally can have a positive effect on both financial and environmental costs when running the system.

This diversity optimization problem can be addressed using submodular functions, which naturally capture the trade-off between relevance and diversity through their diminishing returns property. In the past decades, the use of submodularity in Machine Learning and Natural Language Processing has become increasingly popular. As explained in Bilmes [2022], submodular maximization has been successfully applied to tasks such as extractive summarization of texts and recorded speech, feature selection, and active learning. More recently, submodular optimization has also been used for summarization of input context for LLMs, effectively improving resource efficiency while maintaining performance [Kumari, 2025].

Intuitively, submodular functions can be described as set functions characterized by the property of *diminishing returns*. In broad terms, this means that the marginal gain from adding an element to a set diminishes or remains constant as the set size increases. This fundamental property makes submodular functions particularly well-suited for diversity optimization and information coverage problems. Maximizing a submodular function naturally

leads to the selection of minimally redundant subsets that provide maximum information coverage from the original set within specified constraints.

Formally, one way to define submodular functions is through the *diminishing returns* property, which is often the most intuitive formulation (though equivalent definitions also exist). A set function

$$f : 2^V \to R$$

is called *submodular* if, for all $X, Y \subseteq V$ with $X \subseteq Y$, and for all $v \notin Y$, the following inequality holds:

$$f(X \cup \{v\}) - f(X) \;\geq\; f(Y \cup \{v\}) - f(Y).$$

In this thesis, we explore the application of the optimization of different submodular functions to combine similarity and diversity in the *reranking* step of Retrieval-Augmented Generation. Specifically, we employ **convex combinations** of:

1. A concave transformation (LOG1P, in which $\phi(x) = \log(1 + x)$ [1]) of a feature-based submodular function, combined with a submodular function (Facility Location)
2. A submodular transformation (One-Hot Quantization) of a modular function with manually specified weights for each quantile range, combined with a submodular function (Facility Location)
3. A submodular transformation (One-Hot Quantization) of a modular function with exponentially decaying weights starting from a peak bin and decreasing according to a base decay factor, combined with a submodular function (Facility Location).

A more detailed explanation of the functions can be found in Chapter 4.

We use five existing datasets from different topics (arguments, news, question-answering, and fact-checking) from Zhao et al. [2024] and adapt them to an existing pipeline for RAG

[1] In the greedy algorithm, we require non-negative function values to guarantee that our solution achieves at least $(1-1/e)$ times the optimal value. While $\phi(x) = \log(x)$ does not violate submodularity, it produces negative values when $0 < x < 1$. Therefore, we use $\phi(x) = \log(1 + x)$ instead, which guarantees non-negative output for all $x \geq 0$, thus preserving the approximation guarantee.

evaluation developed by Rau et al. [2024]. We use the golden retrieval documents provided in the datasets and evaluate the precision and recall of the **retrieval** (not the generation) with different variations of hyperparameters. As a baseline, we also run the same experiment but using *Maximal Marginal Relevance* (MMR), one of the well-established and widely-used methods for diversity of information [Goldstein and Carbonell, 1998], instead of submodularity. In addition, we perform a brief error analysis to assess the largest improvements and how they are correlated to linguistic characteristics of the datasets.

Thus, this thesis focuses on answering the following research questions:

1. How do submodular functions perform compared to Maximal Marginal Relevance (an established diversity method) in terms of retrieval precision and accuracy in RAG systems?
2. How do dataset characteristics affect the potential for improvement when using submodular functions to enhance diversity in the retrieval stage?

The contributions are three-fold:

1. The project demonstrates the effectiveness of submodularity in the reranking step of a RAG system to improve the precision and recall of the retrieved documents.
2. The paper shows how to apply submodular optimization to improve diversity while preserving similarity. It shows three combinations of submodular functions and how they can be used to balance similarity and diversity adjusting different hyperparameters.
3. The analysis provides insights into which types of datasets are the ones that benefit the most from submodularity and diversity in general.

This thesis is structured as follows: Chapter 1 provides an Introduction, Chapter 2 presents a Literature Review of work on diverse retrieval for RAG, Chapter 3 outlines the Methodology (including datasets, main pipeline, and evaluation), Chapter 4 presents the Experiments (including a description of the submodular functions implemented), Chapter 5

provides Results and Discussion, and Chapter 6 provides Conclusions and Directions for Future Work.

Chapter 2

LITERATURE SURVEY

This thesis focuses on improving the recall and precision of the *retrieval* component of Retrieval Augmented Generation by incorporating diversity through the application of submodular functions that balance similarity and diversity objectives. Previous work has explored various systems for considering diversity in the retrieval component in RAG systems. As mentioned in the Introduction, submodular functions have been successfully applied in NLP tasks to improve information coverage, since their maximization naturally encourages diversity.

2.1 Diversity in Retrieval Augmented Generation

This work builds on previous efforts to improve diversity in retrieval functions within Retrieval-Augmented Generation (RAG). A range of methods have been explored to address this goal. For instance, in *Open-World Evaluation for Retrieving Diverse Perspectives*, Chen and Choi [2025] run the system with different corpus-retriever pairs and subsequently apply Maximal Marginal Relevance (MMR) reranking to the retrieved results with the goal of boosting diversity. The authors also generate three datasets based on queries that naturally trigger diverse valid perspectives: debate topics and survey questions.

More precisely, the authors' approach experiments with a set of different retrievers, including one sparse retriever (BM25) and four dense retrievers (DPR, Contriever, TART, and NV-Embed-v2). Each retriever is used to obtain the top $R = 100$ documents from the corpus. None of these retrievers considers diversity, as they are solely similarity-based. Subsequently, the authors apply reranking on the set of R retrieved documents through Maximal Marginal Relevance (MMR). MMR is considered one of the oldest and most established methods to

improve diversity, which was introduced for information retrieval and summarization [Goldstein and Carbonell, 1998]. MMR aims to find a balance between relevance and diversity when constructing a subset of documents, selecting documents that are both highly relevant to the query and minimally redundant compared to others already chosen. For a detailed explanation on MMR, which is the baseline for our diversity experiments, please see the Experiment section of this thesis.

The authors evaluate the results using two metrics: MRECALL@5 and PRECISION@5. These are adapted from the standard definitions of recall and precision at cut-off $k = 5$:

$$\text{Recall@5} = \frac{|\{\text{relevant documents in top 5}\}|}{|\{\text{total relevant documents}\}|},$$

$$\text{Precision@5} = \frac{|\{\text{relevant documents in top 5}\}|}{5}.$$

To assess whether a certain document supports a certain perspective or not, the authors build a Perspective Detection model that approximates human judgement. The authors report that although re-ranking with MMR improves diversity (MREC) on half of the retriever-corpus combinations, the precision decreases in all the settings. The rate of improvement in recall ranges from 2.6% to 9.5%. The authors hypothesize that this limited performance could be explained by the fact that semantic dissimilarity does not always translate to differing perspectives.

Another approach to improve retrieval diversity is presented by Li et al. [2024], who introduce the use of Determinantal Point Processes to balance similarity, relevance, and conflicting information in retrieved documents. In *SMART-RAG: Selection using Determinantal Matrices for Augmented Retrieval*, the authors define three key components for context selection. *Textual similarity* is measured via cosine similarity between dense embeddings of contexts, forming a similarity matrix that penalizes selecting similar contexts to maximize diversity. *Conflict relations* are assessed using a Natural Language Inference model to detect contradictory context pairs, producing a conflict matrix that penalizes selecting conflicting contexts to maximize factual consistency. *Query-context relevance* is computed as the cosine

similarity between the query and each context, ensuring selected contexts remain relevant to the query. Using these relations, SMART constructs a conflict-aware kernel matrix for Determinantal Point Process (DPP) based context selection. This kernel matrix incorporates relevance scores and a conflict-adjusted similarity matrix where detected conflicts reduce similarity scores. The SMART model then uses greedy inference to efficiently select a subset of contexts that maximizes determinant gain while avoiding redundancy and improving relevance. The trade-off between relevance and diversity is controlled through hyperparameter tuning.

Li et al. [2024] evaluate their approach on tasks such as question answering, multi-hop reasoning, and fact verification using datasets including NaturalQuestions (NQ), TriviaQA (TQA), and HotpotQA, among others. Their evaluation follows a three-stage pipeline: first, they use Contriever (an unsupervised dense retrieval model by Izacard et al. [2022]) to retrieve the top 50 documents for each query. Next, as *pre-reranking*, they restrict the subset to the top 30 most relevant documents using BGE (an embedding model supporting dense retrieval by Xiao et al. [2024]) based on relevancy scores. Finally, the SMART approach, along with other context-selection methods for comparison, is applied for reranking to select the final set of documents used as input context for text generation. The authors evaluate Exact Match and F1 scores on answers generated using the Llama3-8b-instruct model with the December 2018 Wikipedia dump as the retrieval corpus, following a 5-shot in-context learning setup. The results show that, across all datasets, the SMART approach outperforms baseline methods, including MMR, the standard diversity method. Furthermore, ablation experiments demonstrate optimal performance when the SMART method incorporates all three components (relevance, diversity, and conflict resolution) compared to variations where one or more components are removed. This highlights the value of diversity and conflict-aware selection over purely similarity-based methods.

A different approach to reduce redundancy and improve information content on the retrieved documents is suggested by Pickett et al. [2025]. The authors mention that current RAG systems often retrieve semantically similar but redundant passages, which wastes lim-

ited context window space. With this motivation, they design a new method for retrieval that is based on information gain. This measure computes the total information that is relevant to a query from the retrieved context from the corpus. Although information gain does not explicitly include diversity, the metric organically leans towards diverse subsets of information, as the gain of adding a certain fragment x to a subset A decreases if x is redundant with elements already included in A.

The authors introduce this through *Dartboard*, an algorithm that maximizes relevant information gain. *Dartboard* is based on an analogy of a probabilistic game with two players, in which: 1) **Player 1** picks a hidden target point on a dartboard and throws a dart aiming for it, but the dart lands with some uncertainty (representing the query), 2) **Player 2** is able to see where the dart landed but doesn't know the true target location. Thus, **Player 2** must select k points on the board to minimize the distance between the actual target (representing passage selection) and their closest guess. By rewarding proximity to the original target in **Player 2's** guesses, the system maximizes information coverage without redundancy, while considering closeness to **Player 1's** dart as similarity to the query.

Formally, **Player 1** selects a target T from a set of all points A and gives a query q (their dart). Then **Player 2** makes a set of guesses $G \subseteq A$, resulting in a score $s(G, q, A, \sigma)$, with a distance function D, which is given as:

$$s(G, q, A, \sigma) = \sum_{t \in A} P(T = t \mid q, \sigma) \min_{g \in G} D(t \mid g)$$

After some modifications and an implementation of a Gaussian kernel as distance function, this becomes:

$$s(G, q, A, \sigma) \propto -\sum_{t \in A} \mathcal{N}(q, t, \sigma) \max_{g \in G} \mathcal{N}(t, g, \sigma)$$

The search for the optimal G is then performed through a simple greedy optimization method.

The experiment employs benchmark datasets from Chen et al. [2023] with a question

answering task in which the fragments directly answer the queries, either with a single fragment (simple question answering) or several fragments (information integration). The LLM used in the experiment belongs to the ChatGLM family, but there are no more specifications on its details.

The paper shows positive results, as Dartboard outperformed KNN, MMR, and other baselines on both retrieval metrics and end-to-end QA tasks. The researchers also demonstrated that diversity naturally increases as the uncertainty parameter σ increases, supporting the claim that diversity emerges organically from the information gain optimization.

However, the authors acknowledge that some factors, such as runtime complexity and the need for additional hyperparameter tuning (particularly for the σ parameter), might complicate the implementation of their system on a larger scale.

These previous approaches show diversity is a useful route to improve the performance of Retrieval Augmented Generation. Interestingly, even though some of them, as Determinantal Point Processes [Li et al., 2024] or the Dartboard algorithm [Pickett et al., 2025] implement submodular function intuitions, *submodular functions* are not directly mentioned or systematically applied. As explained by Bilmes [2022], it is not uncommon to find submodular approaches in the literature where there is no explicit mention (and perhaps not even awareness) that the functions being applied are part of the general framework of submodular functions.

To address this gap, we explicitly apply submodular functions to retrieval in retrieval-augmented generation, exploring several functions not previously used in this context.

Chapter 3

METHODOLOGY

For this work, we require a framework that maintains consistency throughout the retrieval, reranking, and evaluation pipeline, while also enabling simple integration of datasets appropriate for diverse retrieval tasks. For this reason, we rely on tools (benchmarking libraries and datasets) developed in previous work.

Specifically, this work employs BERGEN, the Retrieval-Augmented Generation end-to-end benchmarking library introduced by Rau et al. [2024], as the main experimental framework. We modify the structure of five datasets initially adapted by Zhao et al. [2024] for the task of diversity in retrieval, aligning their formats with the BERGEN processing and evaluation pipeline. We extend the BERGEN pipeline to save the necessary components for hyperparameter tuning in submodular optimization. We also implement Maximal Marginal Relevance (MMR), the standard diversity method [Goldstein and Carbonell, 1998], as our baseline for performance comparison. We evaluate precision, recall, and F1 score based on the gold retrieval documents provided in each dataset. Using a 70-30 train-test split, we identify hyperparameters that perform well in worst-case scenarios on samples over the training set, then evaluate these robust configurations on the test set. We do not use a separate development set, as our sampling over the training set provides sufficient validation for hyperparameter selection. These components are described in detail below.

3.1 Pipeline

We utilize most of the pipeline from BERGEN, a comprehensive end-to-end benchmarking library for RAG systems that eases integration of new datasets and experimental configurations. BERGEN is a modularized framework that provides options for each stage of the

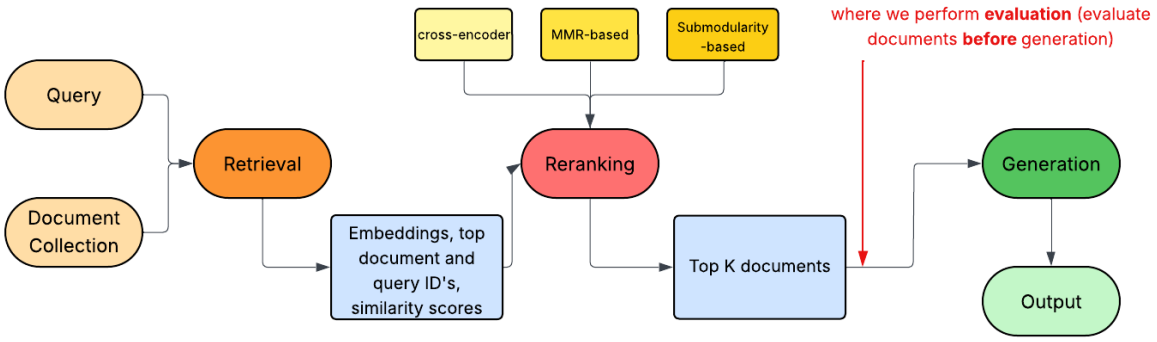

Figure 3.1: General Pipeline for the Experiment

RAG pipeline, including datasets, retriever architectures, rerankers, LLMs for generation, and evaluation metrics.

As previously mentioned, this thesis focuses specifically on evaluating the accuracy and precision of documents retrieved and reranked prior to the generation stage, rather than evaluating the final generated output. This approach allows us to isolate the impact of our submodular diversity optimization on retrieval performance without confounding factors from generation variability. An overview of the experimental pipeline used in our evaluation is illustrated in Figure 3.1.

We compare the performance across two main experimental configurations by evaluating the effectiveness of different reranking approaches: MMR and submodular function-based rerankers. Within the submodular function category, we examine three distinct variants, each with hyperparameter variability that creates different similarity-diversity trade-offs and weighting schemes. This experimental framework allows us to analyze the effectiveness of submodular optimization in comparison to MMR.

As shown in Figure 3.1, both experiments save intermediate results from the retrieval step (document IDs, document embeddings, query IDs, and similarity scores) and apply MMR and submodular functions for reranking, respectively.

Table 3.1: Experimental Setup: Retriever and Reranker Combinations

Config	**Retriever**	**Reranker**	
		Type	**Submodular Function**
1	Contriever	MMR	-
2a	Contriever	Submodular Function	FE using LOG1P + Facility Location (FL)
2b	Contriever	Submodular Function	One-Hot (manual weights) + FL
2c	Contriever	Submodular Function	One-Hot (decay weights) + FL

3.2 Datasets

We experiment with five datasets, originally designed for different (although related) tasks, and reformatted by Zhao et al. [2024] for the purpose of evaluating diversity in retrieval. The paper by Zhao et al. [2024] originally includes six datasets, but because of computational limitations, we did not work with the *Agnews* dataset. The datasets cover domains including *arguments, news, question-answering* and *fact-checking*. Each of the datasets includes: 1) a set of *queries* 2) a *corpus*: a set of candidate documents/passages for retrieval, and 3) a *key reference*: a gold standard mapping between queries and corpus showing the corpus documents that are relevant for each query.

As mentioned, RAG systems benefit from diverse retrieval approaches. Therefore, we selected datasets that naturally benefit from diverse information coverage for our approach. The datasets cover several different domains and perspective types. *Perspectrum* [Chen et al., 2019] focuses on argumentation and stance detection (a claim supports or opposes a given argument). *AmbigQA* [Min et al., 2020] is in the question answering domain and includes questions with multiple valid answers, which reflects ambiguity in interpretation. *AllSides* [Baly et al., 2020] covers news articles and political ideology, contrasting left and right-wing perspectives. *Ex-FEVER* [Ma et al., 2024] addresses fact-checking, with claims supported or refuted by textual evidence. Finally, *StoryAnalogy* [Jiayang et al., 2023] comes from the

narrative understanding domain and explores story similarity, comparing examples based on shared entities or analogical structure. An example from each dataset can be found below in Table 3.2.

Originally, each query in the dataset by Zhao et al. [2024] consists of two components: a *perspective* and a *root query*. A single root query can be combined with multiple perspectives to create different queries; e.g. if the root query is "the presidential elections in Ecuador" and the perspectives are "politically right-leaning article" and "politically left-leaning article", the resulting queries in the dataset will be "politically right-leaning article on the presidential elections in Ecuador" and "politically left-leaning article on the presidential elections in Ecuador." This format presupposes the existence of perspectives in user queries in LLMs, which is not ideal for our goal, as we intend to retrieve diverse perspectives given a general (non-perspective-specific) query. To address this, we transform the five datasets by Zhao et al. [2024] as follows:

1. Separate the perspectives from the root queries (a straightforward step, since the datasets by [Zhao et al., 2024] also include a list of root queries).
2. Create a) a set of perspective-free queries, b) a set of perspectives associated with each query, and c) a mapping that identifies which documents are relevant to a given query and which perspective they represent.

3.3 Evaluation

3.3.1 Data Splitting and Hyperparameter Selection

We employ a 70-30 train-test split for robust evaluation. To select optimal hyperparameters for the submodular implementation, we implement a worst-case performance strategy on the training set. Specifically, we create 1000 random subsets, where each subset contains 30% of the training data. For each hyperparameter configuration, we evaluate its F1 score on all 1000 subsets and select configurations based on their worst-case performance across

Table 3.2: Datasets with Different Perspectives for Multi-Perspective Tasks

Dataset	**Task Description**	**Different Perspectives**
AllSides [Baly et al., 2020]	Find a news article on the topic *terrorism*.	A left-wing news article about terrorism. A right-wing news article about terrorism.
AmbigQA [Min et al., 2020]	What is the legal age of marriage, without parental consent or other authorization?	The legal age of marriage without consent in Nebraska is X. The legal age of marriage without consent in most other states is Y.
Ex-FEVER [Ma et al., 2024]	Mother Teresa was born in Macedonia, a country in Europe where the residents are called Macedonians. Macedonian is a Slavic language.	A claim that supports the sentence. A claim that refutes the sentence.
Perspectrum [Chen et al., 2019]	It should be allowed to have military recruitment in schools.	It should be allowed to have military recruitment in schools. It should not be allowed to have military recruitment in schools.
StoryAnalogy [Jiayang et al., 2023]	Fertilize the soil. Mix seeds into the fertilized soil.	A story with similar entities. A story that serves as an analogy.

these evaluations. For each hyperparameter setting, we record the worst performance across all subsets, simulating potential distribution shifts. We then select the hyperparameter configuration that achieves the best worst-case performance.

3.3.2 Final Evaluation

The selected robust hyperparameters are evaluated on the held-out and unseen test set (which contains 30% of all data) and compared against our MMR baseline using the same train-test split to ensure fair comparison.

3.3.3 Metrics

We evaluate retrieval quality using precision, recall, and F1 score based on the gold standard documents provided for each dataset. Given the structure of our datasets (as explained in Section 3.2.), the gold standard documents for each query encompass all documents corresponding to the different perspectives contained in the dataset. This structure allows us to evaluate diversity through recall metrics, since retrieving more gold standard documents inherently means capturing more diverse perspectives.

Chapter 4

EXPERIMENTS

4.1 Retrieval

4.1.1 Contriever

The first step of our experiment uses Contriever as the document retriever, specifically the MS MARCO fine-tuned version of the pre-trained model. Contriever is a sentence transformer model that generates dense embeddings through mean pooling [Izacard et al., 2022]. The fine-tuned version we use, Contriever-base-msmarco, is particularly effective for semantic search and information retrieval tasks, creating dense vector representations of sentences and documents.

The queries and documents are encoded in the same vector space. Then, document-query similarity is computed using dot product operations. During initial retrieval, Contriever identifies the 512 most relevant documents from the corpus. At this stage, we preserve intermediate outputs (document embeddings and IDs, query IDs, and similarity scores) to enable subsequent application of submodular optimization functions and MMR reranking methods.

4.2 Reranking

4.2.1 Maximal Marginal Relevance (Baseline)

We implement Maximal Marginal Relevance (MMR) as the baseline for comparison. MMR was introduced by Goldstein and Carbonell [1998] as a method to balance query-relevance and information diversity for text retrieval and summarization. This approach provides a linear combination of relevance and novelty, creating *marginal relevance*. A document has

high marginal relevance if it is both highly relevant to the query and minimally redundant with documents already selected.

Formally, MMR is defined as:

$$MMR \stackrel{\text{def}}{=} \arg \max_{D_i \in R \setminus S} \left[\lambda \text{Sim}_1(D_i, Q) - (1 - \lambda) \max_{D_j \in S} \text{Sim}_2(D_i, D_j) \right] \tag{4.1}$$

Here, R is the initial corpus of documents, and $R \setminus S$ is the set difference between R and the subset of already selected documents S. Each next document D_i added to the subset S is selected by maximizing a score that balances two factors: relevance to the query Q and dissimilarity to documents already selected in S. Additionally:

- $\textbf{Sim}_1$ is the retriever similarity score between the candidate document and the query Q, indicating relevance.
- $\textbf{Sim}_2$ is the cosine similarity between document embeddings, capturing redundancy or similarity between documents.
- $\lambda \in [0, 1]$ is a tunable hyperparameter that adjusts the trade-off between maximizing relevance ($\lambda = 1$) and encouraging diversity ($\lambda = 0$).

We apply MMR to rerank the documents for each query using different hyperparameter variations. We normalize the relevance scores, create a similarity matrix to compute document-to-document similarity, and then use a greedy algorithm to iteratively select the document that maximizes MMR.

First, we normalize the relevance scores in three steps:

1. Apply z-score normalization by subtracting the mean and dividing by the standard deviation: $z_i = \frac{x_i - \mu}{\sigma}$
2. Apply the sigmoid function to map values to $[0, 1]$
3. Normalize the scores so they sum to 1

We create the document similarity matrix using cosine similarity:

$$S_{ij} = \frac{x_i \cdot x_j + 1}{2} \tag{4.2}$$

where x_i and x_j are the normalized embedding vectors for documents i and j, respectively. This transformation maps the cosine similarity from the range $[-1, 1]$ to $[0, 1]$.

We then run MMR selection using 11 different values of λ, evenly spaced between 0.9 and 1.0:

$$\lambda \in \{0.90, 0.91, 0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 1.00\} \tag{4.3}$$

We select λ values in this range to prioritize relevance while introducing minimal diversity constraints. Values close to 1.0 mean that retrieved documents are primarily relevant, with only small amounts of redundancy reduction. At $\lambda = 1.0$, the algorithm performs a purely relevance-based ranking, while lower values gradually increase the penalty for selecting redundant documents.

Finally, we save the selected document orders and corresponding hyperparameter values for our evaluation.

4.2.2 Submodular Functions

Submarine: a Submodular Function Library

For the implementation of submodular functions, we rely on *Submarine*, a highly-optimized and highly-scalable discrete optimization suite of tools for machine learning, artificial intelligence, and data science [Bilmes, 2025]. Submarine uses submodular and supermodular functions to improve data efficiency by reducing datasets while preserving their informational content.

Submarine has both command-line interface (CLI) tools and a Python interface. The library includes a wide range of submodular and supermodular functions: calibrated concave functions (including powers, logarithms, mins and soft-mins, exponential mins, etc.), compressed encoded one-hot feature-based functions, facility location functions, set cover functions, graph cut, different log-determinant (e.g. determinantal point processes), among others.

In our implementation, we use Submarine's accelerated greedy algorithm for submodular

maximization, facility location functions, feature-based functions, and one-hot quantization. We also use the library to create a weighted linear combination of different submodular functions to balance the mixing weights between relevance and diversity. Let us describe these in detail below.

1. FE using LOG1P + Facility Location

The first approach implements an objective function that is a convex combination of two submodular functions: a relevance function and a diversity function.

Objective Function

This function is constructed as a **convex combination** (linear combination where coefficients add up to 1) of two submodular functions:

$$f(S) = w_1 \times \text{relevance}(S) + w_2 \times \text{diversity}(S) \tag{4.4}$$

where:

- relevance(S): Feature Extraction function that measures query-document relevance
- diversity(S): Facility Location function that measures document diversity
- $w_1 + w_2 = 1$: Convex combination weights

For the weights, we experiment with 32 different values for the diversity coefficient (w_2) between 0.0 (no diversity) and 0.3 (moderate diversity):

$$\begin{aligned} w_2 \in \{&0, 0.010, 0.019, 0.029, 0.039, 0.048, 0.058, 0.068, 0.077, 0.087, 0.097, \\ &0.106, 0.116, 0.126, 0.135, 0.145, 0.155, 0.165, 0.174, 0.184, 0.194, \\ &0.203, 0.213, 0.223, 0.232, 0.242, 0.252, 0.261, 0.271, 0.281, 0.290, 0.300\} \end{aligned} \tag{4.5}$$

As this is a convex combination, the relevance coefficient w_1 is simply calculated as $1 - w_2$.

Function 1: Relevance Function (Feature Based function using LOG1P)

This is a modular function, which means the marginal gain stays the same independent of the current subset. By using LOG1P, we reduce the skew towards extremely high relevance scores. Formally,

$$\text{relevance}(S) = \sum_{i \in S} \text{CCF}(\text{score}_i) \tag{4.6}$$

where CCF is a **Calibrated Concave Function LOG1P**:

$$\text{CCF}(x) = \log(1 + \gamma \times x) \tag{4.7}$$

Gamma, γ controls the compression of the curve. Lower gamma values mean final values closer to each other, while higher values emphasize differences. We consider the following values for γ:

$$ccf\gamma \in \{0.01, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.5, 1, 5, 10\} \tag{4.8}$$

Function 2: Diversity Function (Facility Location)

A Facility Location (FL) function is used to select the subset that 'covers' the greatest area out of a ground set given certain constraints (as size of the subset or cost) by selecting the points of the set that best represent the whole ground set. It is usually conceived as the solution to the problem of selecting a set of facilities (subset A) in order to represent a set of clients (set U), where each client is represented by their closest facility. In this case, this is equivalent to choosing a subset of documents (subset A) that best represents the full corpus of documents (set U). Formally,

$$\text{FL}(A) = \sum_{u \in U} \max_{a \in A} \text{similarity}(a, u) \tag{4.9}$$

where:

- U: Set of all clients (all documents in the corpus)
- A: Subset of U, selected facilities (documents chosen for the subset)
- similarity(u, a): Similarity between client u and facility a (pair of documents)

Facility location has a number of different parameters that can be adjusted. For instance, we can use different similarity kernels. In the experiments, we use Cosine Similarity, Euclidean Similarity (based on the inverse of Euclidean Distance), and Squared Euclidean Similarity (in which the function is more sensitive to larger similarity distances). Additionally, we consider the hyperparameter of k-Nearest Neighbors Filtering (nnz) to control the number of nearest neighbors each document can have as 'similar'. Lower values mean potentially less diversity, and higher values mean each document has a larger, more sparse range of documents that can be considered its neighbors. We consider the following values for nnz:

$$nnz \in \{8, 10, 16, 24, 32, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130, 140\} \tag{4.10}$$

2. One-Hot Quantization (Manual Weights) + Facility Location

The second approach implements an objective function that is a convex combination of a relevance function and a diversity function.

Objective Function

This works in the same way as ***1. FE using LOG1P + Facility Location*** above, except the Relevance Function is adjusted depending on the quantile of relevance the document falls into.

Function 1: Relevance Function (One-Hot Quantization + Feature Extraction)

The relevance function is similar to the one used in ***1. FE using LOG1P + Facility Location***, but it includes weighting using one-hot quantization with manual bins. This

means that documents are placed in 'bins' according to the range of scores they fall into, and this helps weight their relevance.

$$\text{relevance\_weight} \times \text{CCF}(x) = \text{relevance\_weight} \times \log(1 + \gamma \times x) \tag{4.11}$$

Relevance_weight is calculated using four different hyperparameters. Initially, the documents are divided into three 'bins' (quantiles). The weight of each bin is calculated as follows:

- bin_1 (lower 90% relevance scores) = bottom quantile weight
- bin_2 (medium 5% relevance scores) = ohq multiplier × quantile power
- bin_3 (highest 5% relevance scores) = $\begin{cases} (\text{ohq multiplier} \times \text{quantile power})^2 & \text{if squared} \\ (\text{ohq multiplier} \times \text{quantile power}) \times 2 & \text{if not squared} \end{cases}$

In our experiment, we run the functions with the following values for each hyperparameter:

$$\text{bottom quantile weight} \in \{0.01, 1\} \tag{4.12}$$

$$\text{ohq multiplier} \in \{5, 10, 20\} \tag{4.13}$$

$$\text{quantile power} \in \{2, 3, 4, 5, 6\} \tag{4.14}$$

$$\text{ohq top quantile squared} \in \{\text{true}, \text{false}\} \tag{4.15}$$

This helps fine-tune how much importance we want to give to the score of each document in relation to which bin (quantile) it falls into according to its relevance.

Function 2: Diversity Function (Facility Location)

This function works in the same way as the Facility Location function in ***1. FE using LOG1P + Facility Location***.

3. One-Hot Quantization (Tuple-Based Weights) + Facility Location

The third approach implements an objective function that is a convex combination of a relevance function and a diversity function.

Objective Function

This works in the same way as ***1. FE using LOG1P + Facility Location*** above, except the Relevance Function is adjusted depending on the quantile of relevance the document falls into, and the quantiles are automatically created according to certain hyperparameters.

Function 1: Relevance Function (One-Hot Quantization (Exponential Decay Weights) + Feature Extraction)

The relevance function is similar to the one used in ***1. FE using LOG1P + Facility Location*** and ***2. One-Hot Quantization (Manual Weights) + Facility Location***, but it includes weighting using one-hot quantization with automatically created bins. Documents are placed in 'bins' according to the range of scores they fall into to help weight their relevance, and this bins are weighted using exponential decay from the highest-scoring bin.

$$\text{relevance weight} \times \text{CCF}(x) = \text{relevance weight} \times \log(1 + \gamma \times x) \tag{4.16}$$

In our experiment, documents are divided into 30 bins with exponential weight decay from the center bin (which corresponds to the highest-scoring documents):

$$\text{weight}(\text{bin}_i) = \text{base}^{\max(0,(\text{power\_iterations} - \text{distance\_from\_center\_bin}))} \tag{4.17}$$

The center bin receives the maximum weight of $\text{base}^{\text{power\_iterations}}$, and weights decay exponentially as distance from the center increases. For example, with 30 bins, base=3 and power_iterations=8:

- Bin 29 (highest relevance scores): weight $= 3^8 = 6561$

- Bin 28: weight $= 3^7 = 2187$
- Bin 27: weight $= 3^6 = 729$
- ...
- Bins 21-0 (lowest scores): weight $3^0 = 1$

In our experiments, we test the following hyperparameter values:

$$\text{base} \in \{2, 3, 4, 5\} \tag{4.18}$$

$$\text{number of bins} = 30 \tag{4.19}$$

$$\text{power\_iterations} = 8 \tag{4.20}$$

This approach creates a highly-granular range of weights for the relevance of the documents. It is significantly more complex and nuanced than the initial, non-weighted combination used in ***1. FE using LOG1P + Facility Location***.

Function 2: Diversity Function (Facility Location)

This function works in the same way as Facility Location in ***1. FE using LOG1P + Facility Location***.

Chapter 5

RESULTS AND DISCUSSION

After running experiments for both the baseline (MMR) and our submodular approach, the scripts save the retrieved documents for each hyperparameter combination. Following the methodology described earlier, we then implement a robust hyperparameter selection process: we 1) sample several subsets from the training set and, 2) for each hyperparameter combination, identify its worst F1 performance across these subsets. We then select the hyperparameter combination that achieves the best worst-case performance—essentially choosing the configuration with the highest performance floor. This approach ensures that even under worst-case conditions, our model maintains acceptable performance. We evaluate the selected hyperparameters on the final test dataset. This same robust selection process is applied to both MMR and our submodular approach. The following section presents the precision, recall, and F1 scores for each dataset.

Table 5.1: F1 Score Comparison: MMR vs Submodular (Min Values)

Dataset	**MMR**	**Submodular**	**Improvement (%)**
AllSides	0.3072	0.3072	0.00%
AmbigQA	0.3885	0.3858	-0.69%
Ex-FEVER	0.3072	0.3072	0.00%
Perspectrum	0.6125	0.6138	+0.21%
StoryAnalogy	0.3086	0.3587	+16.23%

The F1 score shows improvement in 2 out of the 5 datasets. Two of the datasets show no difference, and in the case where performance decreases, the decline is not substantial

Table 5.2: Recall Comparison: MMR vs Submodular (Min Values)

Dataset	**MMR**	**Submodular**	**Improvement (%)**
AllSides	0.3063	0.3063	0.00%
AmbigQA	0.4308	0.4256	-1.21%
Ex-FEVER	0.3063	0.3063	0.00%
Perspectrum	0.5797	0.5942	+2.50%
StoryAnalogy	0.3767	0.4700	+24.77%

Table 5.3: Precision Comparison: MMR vs Submodular (Min Values)

Dataset	**MMR**	**Submodular**	**Improvement (%)**
AllSides	0.3081	0.3081	0.00%
AmbigQA	0.3538	0.3528	-0.28%
Ex-FEVER	0.3081	0.3081	0.00%
Perspectrum	0.6493	0.6348	-2.23%
StoryAnalogy	0.2613	0.2900	+10.98%

(about 0.7%). However, there is significant improvement in one dataset, ***StoryAnalogy***, which likely benefits the most from diversity-based selection.

When analyzing the characteristics of StoryAnalogy in comparison to other datasets used in this work, it is noticeable that it includes complex queries that require diverse response types, such as:

Query: "Water flows downwards thanks to gravity. Enters the dam at high pressure."

- **Perspective 1:** A story that is an analogy to the following story → story-corpus-10
- **Perspective 2:** A story with similar entities to the following story → story-corpus-11

Retrieved responses:

- **Story-corpus-10:** "Money flows downwards thanks to market forces. Enter the bank at high interest rates."
- **Story-corpus-11:** "As gravity serenades the cascading waters, they plunge into the expansive dam, compelled by their own force and pressure."

This example demonstrates one of the reasons why submodular optimization might excel on the StoryAnalogy dataset: it successfully captures both analogical transformations (water/gravity => money/market forces) and stylistic variations (technical => poetic descriptions). These are documents that seem much more semantically different, or diverse in comparison to other datasets, which rely more on relevance. For instance, ***Perspectrum***, which shows a much smaller increase in F1 score in comparison to MMR, has gold documents that rely more on semantic similarity to the query:

Query: "It should be allowed to have military recruitment in schools"

- **Perspective 1:** Find a claim that opposes the argument → perspectrum-corpus-0
- **Perspective 2:** Find a claim that supports the argument → perspectrum-corpus-1

- **Perspective 3:** Find a claim that relates to the argument → perspectrum-corpus-2

Retrieved responses:

- **Perspectrum-corpus-0:** "Military recruitment in schools is less education than propaganda"
- **Perspectrum-corpus-1:** "Military recruitment in schools is more propaganda than education."
- **Perspectrum-corpus-2:** "Military recruitment in schools provides more propaganda than it does education."

Thus, the substantial improvement on performance on StoryAnalogy likely is related to the submodular functions providing the diversity needed to satisfy notably different perspective requirements.

When analyzing the recall values, it is clear that the submodular approach either matches or outperforms MMR across most datasets. Only one dataset (AmbigQA) shows a performance decrease of approximately 1.21%. Two datasets, Perspectrum and StoryAnalogy, demonstrate substantial improvements of 2.50% and 24.77% respectively, while AllSides and Ex-FEVER show identical performance between approaches.

In contrast, precision shows more mixed results. Two datasets, Perspectrum and AmbigQA, exhibit decreased precision, while AllSides and Ex-FEVER perform just as well as MMR. StoryAnalogy continues to demonstrate substantial improvement (10.98%), though considerably less than its recall gains.

This divergence between recall and precision performance reflects the fundamental nature of submodular optimization. By diversifying the chosen subset of documents, it metaphorically casts a *wider net*, capturing more relevant documents that might have been overlooked due to their dissimilarity to top-ranked results, while simultaneously becoming more prone to including less precise selections. However, the overall F1 scores demonstrate that submodular optimization generally can improve the performance of Retrieval Augmented Generation systems across these datasets.

Best Performing Hyperparameters

Additionally, we perform a brief analysis of which hyperparameters and functions yield the best precision and recall scores means across all samples in each dataset.

Table 5.4: Best Hyperparameter Configurations for Recall

Dataset	Precision	Recall	nnz	diversity_weight	gamma	ccf	kernel	quantile_power	squared	ohq_binning_spec	ohq_mult
AllSides	0.3704	0.1759	70	0.0194	10	LOG1P	SIMEUCLID	2	False	bins=30,center_bin=29,base=2	5
AmbigQA	0.3623	0.4429	140	0.0097	0.09	LOG1P	SIMEUCLID	3	False	90,1.0;5,15;5,225	5
ex-FEVER	0.3223	0.3424	60	0.0097	0.07	LOG1P	SIMEUCLID	5	True	90,1;5,25;5,625	5
Perspectrum	0.5744	0.7489	110	0.0968	1	LOG1P	SIMEUCLID	6	False	90,0.01;5,120;5,14400	20
StoryAnalogy	0.2544	0.4846	120	0.0968	1	LOG1P	SIMEUCLID	6	False	90,0.01;5,120;5,14400	20

Table 5.5: Best Hyperparameter Configurations for Precision

Dataset	Precision	Recall	nnz	diversity_weight	gamma	ccf	kernel	quantile_power	squared	ohq_binning_spec	ohq_mult
AllSides	0.3778	0.1667	70	0.0097	1	LOG1P	SIMEUCLID	4	False	90,1.0;5,40;5,1600	10
AmbigQA	0.3645	0.4414	60	0.0	0.5	LOG1P	SIMEUCLID	4	True	None	1
ex-FEVER	0.3223	0.3424	32	0.0097	0.08	LOG1P	SIMEUCLID	2	False	90,1.0;5,20;5,40	10
Perspectrum	0.6952	0.6300	110	0.0	0.01	LOG1P	cos	3	False	90,1.0;5,15;5,225	5
StoryAnalogy	0.2766	0.3766	120	0.0968	1	LOG1P	SIMEUCLID	6	False	90,0.01;5,120;5,14400	20

Overall, the best performance predominantly comes from OHQ with manual weights, as evidenced in the *ohq_binning_spec* column where specific percentages of data and their corresponding weights are manually specified (for instance, "90,1.0;5,40;5,1600" for AllSides best precision). This manual weighting approach appears in 8 out of 10 top configurations, with only occasional use of weight exponential decay binning functions or no OHQ at all.

The results consistently demonstrate that all datasets benefit from *some* level of diversity (non-zero *diversity_weight* values), but optimal performance is achieved with relatively modest diversity levels, typically in the lower portion of the tested range (0.0 to 0.0968). The SIMEUCLID kernel, which implements euclidean similarity, dominates the best configurations, appearing in 9 out of 10 cases, with cosine similarity used only once for Perspectrum's best precision score.

Chapter 6

CONCLUSIONS AND FUTURE WORK

This thesis demonstrates that submodular optimization can be an effective approach for improving recall, precision, and F1 scores in the retrieval-reranking stage of Retrieval Augmented Generation systems. This approach proves particularly effective when working with datasets where gold standard documents represent diverse perspectives for each query, and might be less suited to tasks where the gold documents are highly similar to the queries. The submodular approach matches or outperforms MMR (a standard method for diversity-aware retrieval) in most cases. Moreover, the best improvement is about 20% increase in recall in one of the datasets. Even though there are some datasets where the performance is slightly worse, particularly in the precision metric, this is not the general case and is usually a very mild ($\tilde{1}$%) decrease in performance.

Future work could explore implementing submodular functions in RAG systems with datasets from other domains, such as general knowledge tasks, to assess the broader impact of diversity optimization across different contexts. Additionally, other submodular function variants could be investigated to further optimize the similarity-diversity trade-off in retrieval systems.

BIBLIOGRAPHY

Ramy Baly, Giovanni Da San Martino, James Glass, and Preslav Nakov. We can detect your bias: Predicting the political ideology of news articles. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4982–4991, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.404. URL https://aclanthology.org/2020.emnlp-main.404/.

Jeff Bilmes. Submodularity In Machine Learning and Artificial Intelligence, 2022. URL https://arxiv.org/abs/2202.00132.

Jeff Bilmes. Submarine Documentation. https://submarine.page/docs/intro/, 2025. Accessed: August 20, 2025.

Hung-Ting Chen and Eunsol Choi. Open-World Evaluation for Retrieving Diverse Perspectives, 2025. URL https://arxiv.org/abs/2409.18110.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking Large Language Models in Retrieval-Augmented Generation, 2023. URL https://arxiv.org/abs/2309.01431.

Sihao Chen, Daniel Khashabi, Wenpeng Yin, Chris Callison-Burch, and Dan Roth. Seeing things from a different angle:discovering diverse perspectives about claims. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 542–557, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1053. URL https://aclanthology.org/N19-1053/.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. Time-Aware Language Models as Temporal Knowledge Bases. *Transactions of the Association for Computational Linguistics*, 10:257–273, 03 2022. ISSN 2307-387X. doi: 10.1162/tacl$_{a0}$0459.URLhttps://doi.org/10.1162/tacl_a_00459.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-Augmented Generation for Large Language Models: A Survey, 2024. URL https://arxiv.org/abs/2312.10997.

Jade Goldstein and Jaime Carbonell. Summarization: (1) using MMR for diversity- based reranking and (2) evaluating summaries. In *TIPSTER TEXT PROGRAM PHASE III: Proceedings of a Workshop held at Baltimore, Maryland, October 13-15, 1998*, pages 181–195, Baltimore, Maryland, USA, October 1998. Association for Computational Linguistics. doi: 10.3115/1119089.1119120. URL https://aclanthology.org/X98-1025/.

Yufei Guo, Muzhe Guo, Juntao Su, Zhou Yang, Mengqiu Zhu, Hongfei Li, Mengyang Qiu, and Shuo Shuo Liu. Bias in Large Language Models: Origin, Evaluation, and Mitigation, 2024. URL https://arxiv.org/abs/2411.10915.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems*, 43(2):1–55, January 2025. ISSN 1558-2868. doi: 10.1145/3703155. URL http://dx.doi.org/10.1145/3703155.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised Dense Information Retrieval with Contrastive Learning, 2022. URL https://arxiv.org/abs/2112.09118.

Cheng Jiayang, Lin Qiu, Tsz Chan, Tianqing Fang, Weiqi Wang, Chunkit Chan, Dongyu Ru,

Qipeng Guo, Hongming Zhang, Yangqiu Song, Yue Zhang, and Zheng Zhang. StoryAnalogy: Deriving story-level analogies from large language models to unlock analogical understanding. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11518–11537, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.706. URL https://aclanthology.org/2023.emnlp-main.706/.

Lily Kumari. *Submodular Data Selection and Augmentation for Resource-Efficient Learning*. PhD dissertation, University of Washington, 2025. URL https://hdl.handle.net/1773/52976. Department of Electrical and Computer Engineering.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Jiatao Li, Xinyu Hu, and Xiaojun Wan. SMART-RAG: Selection using Determinantal Matrices for Augmented Retrieval, 2024. URL https://arxiv.org/abs/2409.13992.

Huanhuan Ma, Weizhi Xu, Yifan Wei, Liuji Chen, Liang Wang, Qiang Liu, Shu Wu, and Liang Wang. EX-FEVER: A Dataset for Multi-hop Explainable Fact Verification, 2024. URL https://arxiv.org/abs/2310.09754.

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. AmbigQA: Answering ambiguous open-domain questions. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5783–5797, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.466. URL https://aclanthology.org/2020.emnlp-main.466/.

Marc Pickett, Jeremy Hartman, Ayan Kumar Bhowmick, Raquib ul Alam, and Aditya Vempaty. Better RAG using Relevant Information Gain, 2025. URL https://arxiv.org/abs/2407.12101.

David Rau, Hervé Déjean, Nadezhda Chirkova, Thibault Formal, Shuai Wang, Stéphane Clinchant, and Vassilina Nikoulina. BERGEN: A Benchmarking Library for Retrieval-Augmented Generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7640–7663, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.449. URL https://aclanthology.org/2024.findings-emnlp.449/.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. C-Pack: Packed Resources For General Chinese Embeddings, 2024. URL https://arxiv.org/abs/2309.07597.

Xinran Zhao, Tong Chen, Sihao Chen, Hongming Zhang, and Tongshuang Wu. Beyond Relevance: Evaluate and Improve Retrievers on Perspective Awareness, 2024. URL https://arxiv.org/abs/2405.02714.