Multi-aspect Reviewed-item Retrieval

by

Ethan Baron

Supervisor: Scott Sanner

April 2024

B.A.Sc. Thesis

Division of Engineering Science
UNIVERSITY OF TORONTO

Abstract

In this thesis, we explore three extensions to the standard information retrieval setting.

Firstly, in some domains, it is common for users to seek items satisfying multiple independent aspects, expressed through multi-aspect queries. Classical information retrieval algorithms tend to perform poorly on such multi-aspect queries. Previously-proposed methods that explicitly account for multi-aspect structure offer only small improvements in performance over classical methods. We derive a principled algorithm based on a graphical model for multi-aspect retrieval and validate empirically that this algorithm improves performance on multi-aspect queries over classical approaches. Our algorithm offers performance competitive with large language model prompting at a more reasonable computational cost.

Secondly, in many domains, information retrieval systems must leverage vast quantities of user-generated item reviews by effectively combining information about items from multiple user reviews. Past works propose two main branches of algorithms for this reviewed-item retrieval setting, but do not provide theoretical foundations for these algorithms. We formalize reviewed-item retrieval using a graphical model and draw connections between our model and previously-proposed methods, highlighting directions for future research motivated by these theoretical foundations.

Thirdly, we introduce multi-aspect reviewed-item retrieval, which combines the two previous extensions. We show intuitively and formally that multi-aspect queries pose a failure mode for state-of-the-art reviewed-item retrieval methods. We propose a novel reviewed-item retrieval algorithm to address this failure mode, and validate empirically that it achieves improved performance on multi-aspect queries. We also investigate the impact of various design choices for this algorithm.

Acknowledgments

I would like to thank my supervisor, Prof. Scott Sanner, for his guidance and mentorship throughout the course of this project. I have learned many valuable lessons from his words of advice and from the opportunity to participate in the Data-Driven Decision Making Lab this year. These lessons will certainly impact my approach to research in graduate school. I would also like to thank my family, and especially my parents, for their positivity, love, and unconditional support. Neither this thesis nor any of my other achievements would be possible without the energy they instill in me.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Designing effective and scalable information retrieval systems is becoming increasingly important as the quantity of data available on the internet grows rapidly. Beyond the prototypical application of web search engines, information retrieval also plays an important role in various other domains, such as product search engines [1] and conversational recommender systems [2]. In this thesis, we explore three extensions to the standard information retrieval setting which are especially relevant for these alternative domains. The purpose of this thesis is to develop theoretical foundations for these three extensions, and evaluate empirically the performance of principled algorithms derived from these foundations.

1.1 Multi-aspect Retrieval

In some domains it is common for queries to be composed of multiple orthogonal aspects. In these situations, a user is seeking an item that satisfies each of the orthogonal aspects in the query. For example, a user searching for a restaurant may request an "Italian restaurant with a romantic atmosphere", and might not be satisfied by an Italian restaurant with a casual atmosphere or a Chinese restaurant with a romantic atmosphere. Although standard information retrieval algorithms perform poorly on multi-aspect queries [3], this setting, which we call *multi-aspect retrieval*, remains an understudied area in the information retrieval literature.

While Zhang *et al.* [4] show that prompting large language models can improve performance for multi queries, these solutions are computationally expensive and therefore do not scale well. This motivates the need for a more principled approach to boost performance for multi-aspect retrieval at a lower computational complexity than prompting LLMs. They

also suggest several methods that explicitly separate the query into its distinct aspects and then aggregate across aspects to arrive at item-level scores. They argue that this explicit aspect-based approach can help improve explainability and verifiability of information retrieval systems faced with multi-aspect queries. Unfortunately, when evaluated empirically, the aspect-based strategies they investigate offer only minor improvements in performance over standard methods.

We develop a graphical model to represent multi-aspect retrieval, and use this model to derive a principled algorithm for multi-aspect retrieval which is related to the methods proposed by Zhang *et al.* [4]. We validate our algorithm empirically, showing that this improves upon standard information retrieval approaches for multi-aspect queries.

1.2 Reviewed-item Retrieval

User-generated reviews are becoming ubiquitous across many domains. For example, vast amounts of reviews on restaurants, movies, and consumer products help fuel search engines for various popular websites [5]. These reviews offer an important input source to information retrieval systems, but the problem of aggregating information across reviews to arrive at an item-level ranking has also been under-studied in the information retrieval literature. This extension is known as the *reviewed-item retrieval* setting [6].

There are two main frameworks for reviewed-item retrieval in the literature, known as early fusion and late fusion [7]. Although there is limited empirical research on this subject, the late fusion approach appears to perform more strongly in practice. However, past works on reviewed-item retrieval are experimental in nature, and the theoretical justifications for early fusion and late fusion have not been explored.

We develop a graphical model for reviewed-item retrieval, and use this model to develop theoretical justification for early fusion and late fusion, motivating specific directions for future research.

1.3 Multi-aspect Reviewed-item Retrieval

We consider for the first time the combination of the previous two extensions. That is, we consider handling multi-aspect queries in the reviewed-item retrieval case, which we call the *multi-aspect reviewed-item retrieval* setting.

A related line of work is on *aspect-based sentiment summarization*, which involves de-

signing systems to automatically extract aspects and aspect-level scores from a set of item reviews [8]. In contrast to these systems, the multi-aspect reviewed-item retrieval problem requires scoring items against aspects provided in a specific query, rather than automatically extracting aspects and sentiment scores from the reviews. This multi-aspect reviewed-item retrieval problem has never been studied specifically.

We develop a graphical model for multi-aspect reviewed-item retrieval, and provide an argument that multi-aspect queries are a failure mode for existing reviewed-item retrieval methods. To account for this failure mode, we propose a principled algorithm for multi-aspect reviewed-item retrieval called aspect-based late fusion. We verify empirically that aspect-based late fusion offers improved performance on multi-aspect queries compared to standard reviewed-item retrieval algorithms.

1.4 Outline

The structure of the thesis is as follows. We present a review of relevant literature in Chapter 2, establishing the motivation and context for our contributions. In Chapter 3, we provide graphical models for the three settings we explore and develop principled algorithms based on these graphical models. In Chapter 4, we provide details on our implementation and evaluation of these algorithms in practice, including an introduction of the two datasets we use. In Chapter 5, we present the results of our empirical evaluation and discuss the main findings. We conclude in Chapter 6 with a discussion of possible future research directions based on our contributions.

Chapter 2

Background

In this chapter, we present the background for our research, including a discussion of relevant literature. We begin by providing a general introduction to information retrieval. We then introduce the main varieties of information retrieval algorithms, highlighting the recent use of large language models for information retrieval. Lastly, we describe the multi-aspect retrieval and reviewed-item retrieval settings, and summarize previous works on these tasks.

2.1 Overview of Information Retrieval

Information retrieval systems are designed to rank the most relevant items from a set $S = \{s_1, s_2, \ldots, s_n\}$ to a user's information need based on a given query q. In typical applications such as internet search engines, items are natural language documents such as web pages.

It is logical that an information retrieval system would rank candidate items according to decreasing probability of being relevant to the user, given the user's query. This idea is known as the *probability ranking principle*, and Robertson [9] shows that under two conditions, this principle indeed maximizes the effectiveness of the system. The first condition, known as the *binary relevance* assumption, means each item is either relevant to the user's information need or not. The second condition is that the relevance of an item to a given query is independent of the other candidate items.

We present a graphical model representing these assumption in Figure 2.1. Here, q refers to the query, s_i refers to the i^{th} document, and r_i is a binary variable indicating whether

document i is relevant to the query q.

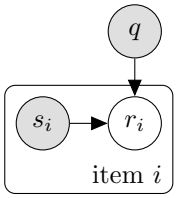

Figure 2.1: Bayesian network representing information retrieval.

This graphical model allows us to derive principled algorithms to optimize various objectives. We focus on the expected 1-call@k objective, which is the probability that at least one of the first k ranked items is relevant. Formally, we denote this as:

$$R_k = \bigvee_{j=1}^{k} r_k \tag{2.1}$$

As in Sanner *et al.* [10], we consider a greedy approach to optimizing this objective. Suppose we have already chosen the first $k-1$ documents, $S_{k-1} = \{s_1, s_2, \ldots, s_{k-1}\}$. We now want to pick the document s_k which maximizes $P(R_k|q, S_{k-1}, s_k)$. Due to independence between items, we have:

$$\underset{s_k \in S \setminus S_{k-1}}{\arg\max}\ P(R_k|q, S_{k-1}, s_k) = \underset{s_k \in S \setminus S_{k-1}}{\arg\max}\ P(r_k|q, s_k) \tag{2.2}$$

Equation 2.2 implies that to optimize expected 1-call@k under our model, one must estimate the probability of relevance $P(r_k|q, s_k)$ for each candidate item, and then pick the item with the highest such probability. Indeed, this result is consistent with the probability ranking principle.

Note that the assumption of independence between items may not always hold true in practice. For instance, if two candidate items are very similar, then their relevance probabilities are likely to be strongly correlated. This observation has given rise to diversification algorithms for information retrieval, which seek to provide users with a ranking that includes a diverse set of items which address a variety of possible intents. However, in this thesis, we maintain the assumption of independence between items, and therefore do not require such diversification techniques. Instead, we focus on methods to estimate relevance scores for each item, and assume that the probability ranking principle is applied to rank the items in decreasing predicted relevance.

2.2 Evaluation of Information Retrieval Systems

To evaluate an information retrieval system, one must measure how effectively that system identifies items that are relevant to a user's information need based on a given query. In this thesis, we focus on cases where retrieved items are presented to the user as an ordered list, and therefore consider methods to evaluate rankings of items.

The standard approach to quantitatively evaluate information retrieval systems is by using test collections. These test collections include a set of manually-labelled relevance judgements between items and queries. These relevance judgements are used to evaluate the ranking of items produced by a system for each query in the collection. A wide variety of metrics can be used for this evaluation, such as *mean average precision* (MAP), *R-precision*, and *mean reciprocal rank* (MRR) [11].

In this thesis, we evaluate our methods using two such test collections. We discuss our choice of evaluation metrics in Section 4.3 and explain why the chosen metrics are well-suited to our tasks.

2.3 Standard Information Retrieval Algorithms

A common approach in information retrieval is the vector space model. In this approach, queries and documents are represented in the same vector space, and documents are scored based on their similarity to the query $\text{Sim}(q, s_i)$. Traditionally, these vector representations are based on term frequency vectors indicating the relative frequency of each word within a given document, giving greater importance to words that appear multiple times in that document. Since these term frequency vectors usually include only a small proportion of non-zero entries, techniques which are based on these representations are known as *sparse retrieval* techniques.

One common sparse retrieval approach is to use *term frequency-inverse document frequency* (TF-IDF) representations, which downweight the term frequency vector based on the document frequency of each word, so words that appear across many documents are considered less important than rarer words. Cosine similarity is commonly used as the similarity scoring function [12].

More recently, natural language encoders such as BERT [13] are used to obtain dense vector representations of queries and documents in the same embedding space. As with sparse retrieval, a similarity function such as cosine similarity between the query embedding

and document embeddings is used to score the documents by relevance. This approach is known as *dense retrieval*, and has been demonstrated to improve upon classical methods in some settings [14].

2.4 Large Language Model Prompting

Recent advancements in natural language processing have created extremely powerful large language models capable of reasoning about human language at unprecedented levels. These advancements were primarily spurred by the introduction of the transformer architecture, which uses a mechanism called *attention* to better understand the context of natural language [15]. Models based on this architecture, such as GPT-3, have displayed state-of-the-art off-the-shelf performance on a variety of tasks, including question answering, translation, reading comprehension and sentence completion [16].

These results were achieved with a method known as *in-context learning*. For in-context learning, the large language model is provided a prompt providing instructions for a given task and, optionally, one or more examples of the task. By appending to the prompt another input for the task, the large language model will automatically generate an output based on the provided instructions. If several examples of the task are provided as part of the prompt, this strategy is known as *few-shot prompting*. If only the instructions are provided without any examples, this is known as *zero-shot prompting* [16].

The success of large language model prompting on various natural language tasks has motivated research on the use of large language model prompting for ranking items in information retrieval. Indeed, it has been shown that passing documents and queries as input prompts to a large language model can improve performance of information retrieval systems. For example, Nogueira *et al.* [17] fine-tune a sequence-to-sequence architecture based on T5 [18] to produce the tokens "true" or "false" given the query and document as the input prompt. They apply a softmax to the logits for these two tokens to obtain an estimate for the probability of relevance. They show that this technique yields state-of-the-art performance on multiple information retrieval datasets. Since this method computes a relevance score for one item at a time, it is considered to be a *pointwise ranking* method.

In contrast, Ma *et al.* [19] propose a *listwise ranking* approach. Specifically, they design prompts that include multiple passages and ask the large language model to sort the passages in order of their relevance to the query. They show that listwise ranking using GPT-3 [16] achieves superior performance to pointwise ranking on several datasets.

Unfortunately, due to the relatively high computational cost of using generative large language models, it is generally infeasible to use these models to predict the probability of relevance for each of the candidate items. This problem is compounded for listwise ranking, which could require passing a large input context to the model composed of multiple entire documents.

Therefore, in practice, researchers have proposed multi-stage information retrieval pipelines [20]. In the first stage, a classical retrieval algorithm is used to rank a subset of candidate items which are likely to be relevant. In subsequent stages, known as *reranking* stages, this subset is re-ordered, and possibly filtered down further, to identify the top few items. By ensuring that the subset passed into a reranking stage is small, using a large language model for reranking becomes computationally manageable.

2.5 Textual Entailment

Textual entailment, also known as natural language inference, describes a relation whereby a given input text implies that a certain hypothesis is true. Entailment models can be used to compute the probability that a given input sequence entails a given hypothesis. For example, an entailment model might predict that the probability that the sequence "bacon cheeseburger" entails the label "delicious" is 0.93, whereas the probability that this sequence entails the label "vegetarian" is 0.02.

Clinchant *et al.* [21] were the first to propose that textual entailment may be a well-suited way to evaluate relevance in information retrieval. In particular, they suggest ranking documents based on their probabilities of entailing the query, and show that this approach yields an improvement over simply computing similarity scores between documents and queries.

Classically, entailment models used term-frequency representations in the vein of sparse retrieval techniques. More recently, large language models such as BERT have been fine-tuned on curated natural language inference datasets to achieve state-of-the-art performance on textual entailment tasks [13]. To the best of our knowledge, such pre-trained entailment models have not yet been applied in information retrieval. Although inference with these models remains more computationally expensive than performing sparse or dense retrieval, they may offer a more scalable alternative to prompting a generative model such as GPT-3.

2.6 Multi-aspect Retrieval

In many domains, queries can involve multiple orthogonal aspects. For example, a user searching for a restaurant with "good sushi and live music" would only be satisfied by restaurant recommendations that offer both sushi and live music. We call the problem of handling such queries *multi-aspect retrieval*. Existing approaches based on the vector space model can be inadequate at capturing these multiple aspects [3].

Zhang *et al.* [4] develop a dataset of queries for recipes satisfying multiple aspects. For each query, the dataset includes a labelled correct recipe title and a set of hard negatives which fail to satisfy all aspects. They test a variety of algorithms on this dataset, finding that generally large language model prompting performs best, followed by dense retrieval, followed by sparse retrieval. In contrast with standard techniques, which they call "monolithic", they also propose aspect-based variations of these algorithms where items are scored against each of the aspects in the query before being aggregated to obtain a final item-level score. These aspect-based algorithms had mixed effects on performance relative to their monolithic counterparts.

Clarke *et al.* [22] present a theoretical view of information retrieval which can be considered to be multi-aspect. Specifically, they model the user's information need as a set of *information nuggets*, and that each document contains information related to a set of information nuggets. They define the probability of relevance as the probability that a document contains at least one information nugget in the user's information need. They assume that the probability of each nugget being in the user's information need or in a given document are independent and present a method to estimate these based on human relevance assessments.

Kong *et al.* [3] extend dense retrieval for the multi-aspect case by explicitly learning separate vector embeddings for each aspect of a query or document. Specifically, they adapt BERT and attention to design an aspect extraction network that explicitly learns separate vector embeddings for each aspect of a query or document. They then include a step to fuse all the aspect embeddings of the query or document and then perform the classic dense retrieval scoring on these fused embeddings. Although this method successfully accounts for multi-aspect queries, one limitation is that their method requires *a priori* knowledge of the set of possible aspect categories. Unfortunately, in some domains, such as conversational recommendation of recipes, the set of possible aspect categories is not known in advance, limiting the usefulness of their approach [4].

2.7 Reviewed-item Retrieval

Another interesting extension to information retrieval is the case of reviewed-item retrieval, where each item s_i is related to a set of multiple documents $\{d_{i1}, d_{i2}, \ldots, d_{im}\}$ [6]. For example, rather than having information about a restaurant contained in one document, the information about this restaurant may be available in a set of customer reviews about the restaurant. In order to rank items, information across multiple reviews must first be fused together to reach an item-level score. Zhang *et al.* [7] describe two general approaches to this fusion problem, which they call *early fusion* and *late fusion*. For a diagram comparing these two approaches, see Abdollah Pour *et al.* [6].

The early fusion approach involves creating an item-level representation from each item's reviews, and then scoring this representation against the query to obtain an item-level score. For instance, in a sparse retrieval setting, the term frequencies from each of the reviews can be added together, forming an item-level term frequency vector [7]. Alternatively, in a dense retrieval setting, the item embeddings can be set to the average of the review embeddings [6]. One benefit of this early fusion approach is that item-level aggregation can be performed in advance of accepting any queries, potentially reducing the need to store all reviews in memory.

In contrast, the late fusion approach involves scoring each review against the query separately, and then aggregating these review-level scores to obtain an item-level score. Zhang *et al.* [7] suggest using top-K aggregation for late fusion, where the item-level score is computed as the sum of the K highest review-level scores. The review-level scores can be based on a standard information retrieval approach, such as sparse retrieval or dense retrieval.

There has been some empirical research of reviewed-item retrieval. Zhang *et al.* [7] evaluate early fusion and late fusion methods with sparse retrieval on two datasets with a similar multi-level structure to reviewed-item retrieval. Early fusion and late fusion each performed best on one of the two datasets, suggesting that both approaches can be competitive for reviewed-item retrieval. However, Abdollah Pour *et al.* [6] consider early fusion and late fusion techniques on a restaurant reviewed-item retrieval dataset, and find that late fusion consistently outperforms early fusion. Similarly, Bursztyn *et al.* [23] use late fusion with $K = 1$ for ranking on another restaurant reviews dataset. We therefore consider late fusion to be the state-of-the-art approach for reviewed-item retrieval.

Chapter 3

Models

In this chapter, we extend the basic graphical model for information retrieval to account for the multi-aspect retrieval, reviewed-item retrieval, and multi-aspect reviewed-item retrieval settings. For each setting, we describe how the graphical model gives rise to a principled retrieval algorithm for that setting, and discuss connections between the theoretical framework and previously-proposed methods. Lastly, we justify our use of a pre-trained textual entailment model to estimate the probabilities arising in our proposed algorithms.

3.1 Multi-aspect Retrieval

We present a graphical model for multi-aspect retrieval in Figure 3.1. In contrast to the standard information retrieval case in Figure 2.1, the query is now represented as a set of aspects indexed by j. We also introduce binary variables c_{ij}, indicating whether item i addresses aspect j. Note that this model assumes that the query aspects q_j are observed directly.

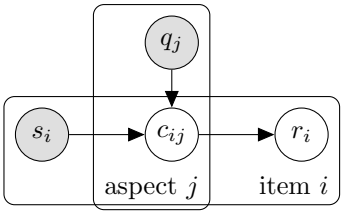

Figure 3.1: Bayesian network representing multi-aspect retrieval.

Under this graphical model, adapting the greedy optimization objective in Equation 2.2

to the multi-aspect retrieval setting yields:

$$P\left(r_i|q_j, s_i\right) = \sum_{c_{ij}} P\left(r_i|c_{ij}\right) P(c_{ij}|s_i, q_j) \tag{3.1}$$

In this thesis, we assume that an item which is relevant to the multi-aspect query should cover all aspects in that query. This is in contrast to the assumption in Clarke *et al.* [22] that a relevant item must only satisfy one information nugget in the user's information need. Formally, we define the conditional probability as follows:

$$P\left(r_i|c_{ij}\right) = \bigwedge_i c_{ij} \tag{3.2}$$

Due to this definition, the only nonzero term in the summation in Equation 3.1 is the term where $c_{ij} = 1$ for all j. This therefore gives the following objective, which factorizes over the query aspects:

$$P\left(r_i|q_j, s_i\right) = \prod_j P(c_{ij} = 1|s_i, q_j) \tag{3.3}$$

We have therefore shown that under the graphical model in Figure 3.1 and the definition in Equation 3.2, the probability of relevance can be estimated as the product of the probabilities of satisfying each of the aspects in the query.

This algorithm is related to the aspect-based approaches taken in Zhang *et al.* [4]. However, they do not focus on estimating aspect coverage probabilities, and instead use various similarity functions (such as sparse or dense retrieval algorithms) to compute scores for each aspect. They then compare a variety of aggregation methods to combine scores across the aspects, such as taking the maximum score, minimum score, arithmetic mean, or geometric mean. We note that if aspect coverage probabilities were used instead of arbitrary similarity scores, the geometric mean aggregation they use would result in an identical ranking to the product aggregation we propose.

3.2 Reviewed-item Retrieval

We present a graphical model for the reviewed-item retrieval case in Figure 3.2. Here, k indexes the reviews for a given item, such that d_{ik} corresponds to review k of item i. We assume that the set of reviews $d_i = \{d_{i1}, d_{i2}, \ldots, d_{im}\}$ are independent and identically distributed, given the item s_i. This reflects the intuition that each review is a representation

of one user's experience with the item, and these experiences can be treated as independent and identically distributed.

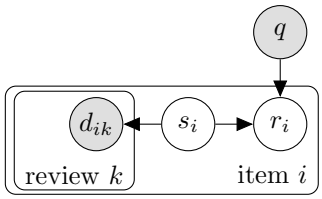

Figure 3.2: Bayesian network representing reviewed-item retrieval.

Note that unlike in the standard information retrieval case, the item s_i is not observed directly. Instead, we must make inferences about the item's relevance based only on the information available in the reviews d_{ik}. Specifically, if we assume that s_i is represented in some continuous space, then the probability of relevance can be calculated via Bayesian marginalization as:

$$P(r_i|q, d_i) = \int_{s_i} P(r_i|q, s_i) \prod_k P(d_{ik}|s_i)\, ds_i \tag{3.4}$$

The early fusion algorithms described by Zhang *et al.* [7] and Abdollah Pour *et al.* [6] can be interpreted as *maximum a posteriori* estimates of this expression using a point estimate for s_i. For example, if review embeddings are assumed to be normally-distributed about some latent item embedding, then the average early fusion approach in [6] is equivalent to applying the maximum likelihood estimate of $P(s_i|d_i)$. Equation 3.4 motivates a Bayesian approach to early fusion, which has not yet been explored.

In practice, it may be challenging to develop an appropriate generative model for $P(d_{ik}|s_i)$, making this early fusion approach challenging. Instead, it may be more feasible to construct a scoring function f such that $\mathbb{E}[f(d_{ik}, q)] = P(r_i|q, s_i)$. The probability of relevance for each item can then be estimated by applying the scoring function directly to each review d_{ik}. We call this approach *naive late fusion*, and it yields the following estimator:

$$\theta_{\text{NLF}} = \frac{1}{m} \sum_{k=1}^{m} f(d_{ik}, q) \tag{3.5}$$

Naive late fusion is a special case of the late fusion approach described by Zhang *et al.* [7]. In their model, the scoring function need not satisfy $\mathbb{E}[f(d_{ik}, q)] = P(r_i|q, s_i)$. Furthermore, in addition to using the average of the review-level scores $f(d_{ik}, q)$, they also consider using top-K aggregation with $K < m$, which takes the sum of the top K review scores for each item.

3.3 Multi-aspect Reviewed-item Retrieval

Our graphical model for multi-aspect reviewed-item retrieval combines the extensions described in the graphical models for multi-aspect retrieval and reviewed-item retrieval. This combined graphical model is presented in Figure 3.3.

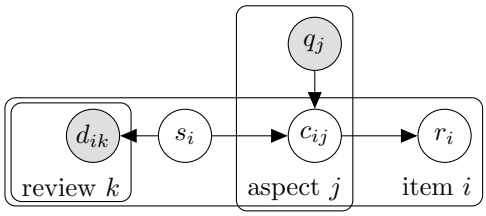

Figure 3.3: Bayesian network representing multi-aspect reviewed-item retrieval.

Under this graphical model, we have:

$$\begin{aligned} P(r_i|q_j, d_i) &= \sum_{c_{ij}} \int_{s_i} P(r_i|c_{ij}) \prod_j P(c_{ij}|q_j, s_i) \prod_k P(d_{ik}|s_i)\, ds_i \\ &= \int_{s_i} \prod_j P(c_{ij} = 1|q_j, s_i) \prod_k P(d_{ik}|s_i)\, ds_i \qquad \text{due to Equation 3.2} \end{aligned}$$

This corresponds to a multi-aspect view of the early fusion approach in Equation 3.4.

Typical late fusion approaches, including the naive late fusion approach from Equation 3.5, can be applied to this setting by simply ignoring the multi-aspect structure of the query. In Figure 3.4, we show a toy example illustrating that the standard late fusion approach can fail for multi-aspect queries. Note that in this example, only the first candidate item has reviews that address both aspects of the query. In spite of this, the aggregate score for this item is lower than that of the third item, which only includes reviews that address one aspect of the query. Naive late fusion has therefore failed to detect the item which best satisfies the multi-aspect query.

To address this failure mode, we propose a variation of naive late fusion, which we call *aspect-based late fusion*. Aspect-based late fusion applies the following algorithm:

1. Each review is scored against each aspect separately using a scoring function g
2. Scores are combined across reviews to arrive at an item-level score for each aspect
3. Scores are aggregated across aspects to arrive at an item-level relevance score, which is used for ranking

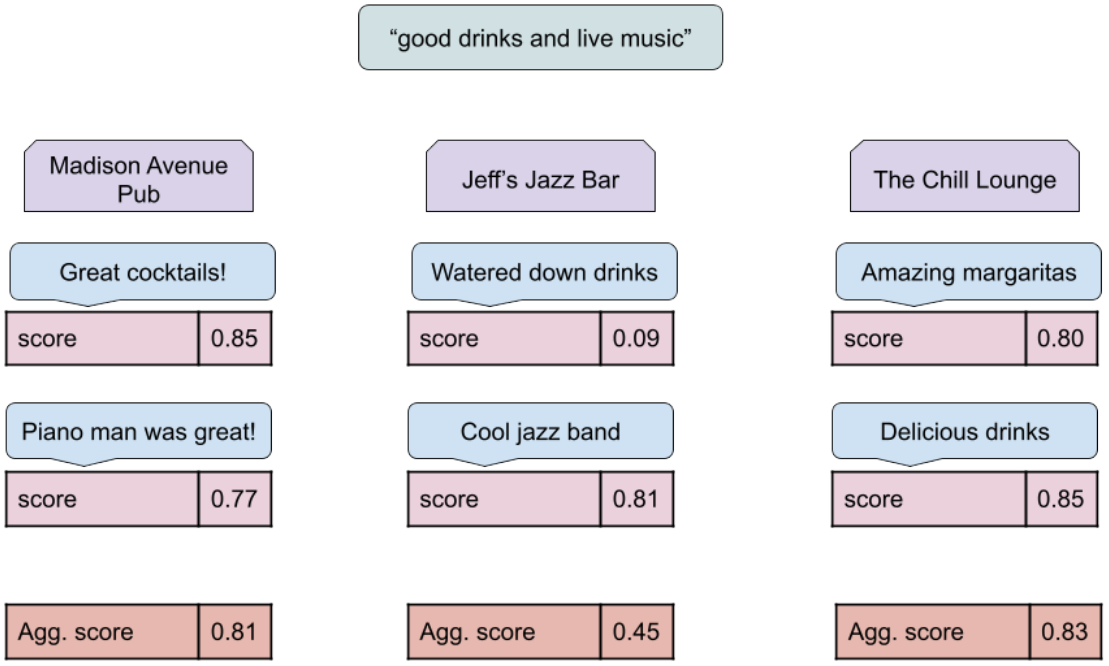

Figure 3.4: Example of multi-aspect failure mode for naive late fusion

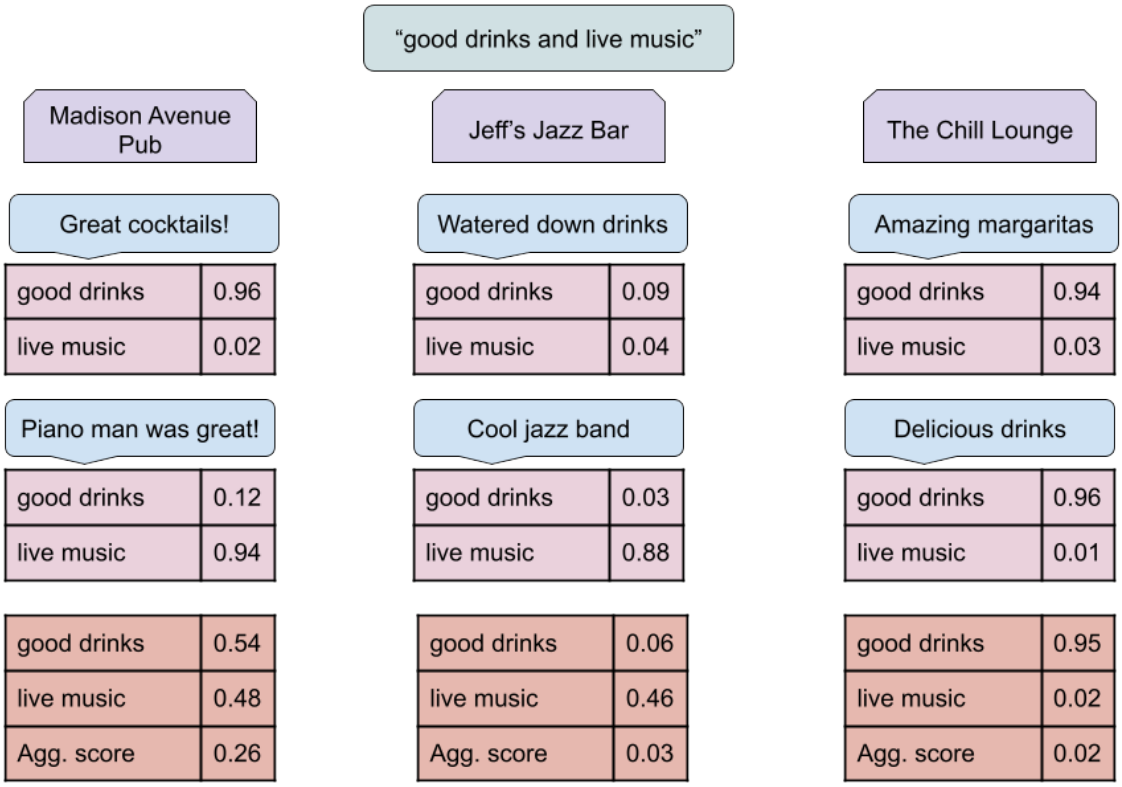

Figure 3.5: Example of aspect-based late fusion

Intuitively, by explicitly modelling the relevance of the item with respect to each of the query aspects, aspect-based late fusion can better detect cases where an item fails to satisfy one or more of the aspects, and correctly gives this item a lower ranking. In Figure 3.5, we adapt our toy example to illustrate aspect-based late fusion, following the three steps above. For this toy example, the final aggregate score for the first item now correctly reflects that it best satisfies the original multi-aspect query. Additionally, the aspect-based late fusion algorithm can improve explainability for rankings by highlighting which reviews gave high scores for each of the aspects in the query.

Concretely, aspect-based late fusion computes the following estimate:

$$\theta_{\text{ABLF}} = \prod_{j} \left(\frac{1}{m} \sum_{k=1}^{m} g(d_{ik}, q_j) \right) \tag{3.6}$$

This algorithm can be viewed as a principled estimate of the true probability of relevance under some conditions. Specifically, if we assume that the scoring function g gives unbiased estimates of the true aspect coverage probability, such that $\mathbb{E}[g(d_{ik}, q_j)] = P(c_{ij} = 1|q_j, s_i)$, then each aspect coverage probability can be estimated as the mean of the review-level scores. In turn, the probability of relevance can be estimated as the product of the aspect coverage probabilities.

In Appendix A, we prove that under certain assumptions, aspect-based late fusion and naive late fusion both offer unbiased estimates of the true relevance, but aspect-based late fusion offers a lower-variance estimate. This provides a theoretical justification suggesting that aspect-based late fusion is a more desirable algorithm for multi-aspect reviewed-item retrieval than naive late fusion.

3.4 Use of Pre-trained Entailment Models

Our proposed algorithm for multi-aspect retrieval, whose objective is presented in Equation 3.1, relies on estimates of the probabilities $P(c_{ij} = 1|s_i, q_j)$. We consider this task as equivalent to evaluating the probability $P(r_i = 1|s_i, q = q_j)$ in the standard single-aspect information retrieval setting. Although classic information retrievals such as sparse retrieval and dense retrieval may provide useful relevance scores, these methods are not designed specifically to generate probabilistic estimates of relevance. While we do consider using the scores produced by these methods in place of the probabilities $P(c_{ij} = 1|s_i, q_j)$ in practice, we hypothesize that an alternative solution designed to give probabilistic outputs may be

more appropriate.

Large language model prompting offers one way to obtain well-calibrated relevance probabilities [17]. However, these techniques are computationally expensive and therefore not tractable at scale. For this reason, we instead propose leveraging a pre-trained entailment model to estimate the relevance probabilities.

Specifically, we propose using a pre-trained entailment model to compute the probability that a given document entails a given query aspect. We hypothesize that these entailment probabilities will provide useful estimates of $P(c_{ij} = 1|s_i, q_j)$ for our multi-aspect retrieval algorithm. Similarly, we hypothesize that their well-calibrated probabilistic outputs are natural choices for the scores $g(d_{ik}, q_j)$ in aspect-based late fusion. Although classical textual entailment models have been applied to information retrieval [21], our use of a pre-trained large language entailment model for information retrieval is a novel contribution.

Chapter 4

Empirical Evaluation

In this chapter, we describe our implementation of the proposed algorithms and how we evaluate these algorithms using two real-world information retrieval datasets. Our experiments are designed to answer the following research questions:

RQ1. Are pre-trained textual entailment models competitive rankers for multi-aspect queries?

RQ2. Which aggregation methods yield the best performance for aspect-based fusion with multi-aspect queries?

RQ3. How does the performance of retrieval algorithms vary for multi-aspect queries with particular attributes?

RQ4. Are pre-trained textual entailment models competitive scoring functions for naive late fusion with multi-aspect queries?

RQ5. How do the choices of scoring function, aggregation method, and hyperparameter K impact the performance of aspect-based late fusion for multi-aspect reviewed-item retrieval?

RQ6. What are potential failure modes for aspect-based late fusion?

We proceed by describing the design decisions in our algorithm implementations. We then introduce the two datasets used for our experiments. Lastly, we present the evaluation criteria we use to compare the performance of different algorithms.

4.1 Implementation Details

To evaluate the performance of pre-trained textual entailment models, we use a version of the BART autoencoder [24] which has been trained on the Multi-Genre Natural Language Inference dataset [25]. This model is provided by Hugging Face[1] and includes 407 million parameters.

As a baseline, we evaluate the performance of dense retrieval using a version of the DistilBERT encoder [26] which has been fine-tuned on the MS MARCO passage retrieval dataset [27] using balanced topic aware sampling, as described by Hofstätter *et al.* [28]. This architecture, known as TAS-B, achieved state-of-the-art results on several information retrieval datasets when compared against other algorithms with similar computation complexity. The fine-tuned model includes 66.4 million parameters and is also provided by Hugging Face[2].

For aspect-based fusion, we consider four aggregation methods, closely related to the four aggregation methods considered in Zhang *et al.* [4]. These include using the maximum score ("Max"), the minimum score ("Min"), the average score ("Mean"), and the product of the scores ("Product"). Of these four aggregation methods, the "Product" case corresponds to view of aspect-based fusion given in Equations 3.3 and 3.6. In addition to these aspect-based methods, we also consider the standard approach of scoring items against the original query without explicitly considering its multi-aspect nature. We call this the "Naive" method.

Since it has been shown that using top-K aggregation can improve performance for late fusion methods in reviewed-item retrieval [6], we consider various settings of the hyperparameter K in our experiments. The case where K is equal to the number of reviews m corresponds to the algorithms in Equations 3.5 and 3.6.

Our implementations are made available open-source on GitHub[3].

4.2 Datasets

We use two open-source datasets to answer our research questions: RecipeMPR[4] ([4]) and RIRD[5] ([6]).

[1] `https://huggingface.co/facebook/bart-large-mnli`
[2] `https://huggingface.co/sebastian-hofstaetter/distilbert-dot-tas_b-b256-msmarco`
[3] `https://github.com/baronet2/marir`
[4] `https://github.com/D3Mlab/Recipe-MPR`
[5] `https://github.com/D3Mlab/rir`

4.2.1 Dataset 1: RecipeMPR

The RecipeMPR dataset includes 500 examples of queries for recipes satisfying multiple aspects. For each query, the dataset lists five candidate recipe titles, each of which satisfy at least one aspect in the query. For each query, only one candidate item satisfies all of the aspects, and this item is labelled as the correct option. This setup aligns closely with the assumption in Equation 3.2, so this dataset therefore offers an ideal setting in which to test our aspect-based approach.

As an example, one query in the RecipeMPR dataset is "I would like a meat lasagna but I'm watching my weight". The aspects annotated for this query are "meat lasagna" and "watching my weight". The dataset provides the following candidate items for this query:

- "Vegetarian lasagna with mushrooms, mixed vegetables, textured vegetable protein, and meat replacement"
- "Forgot the Meat Lasagna with onions, mushrooms and spinach"
- "Beef lasagna with whole-wheat noodles, low-fat cottage cheese, and part-skim mozzarella cheese"
- "Cheesy lasagna with Italian sausage, mushrooms, and 8 types of cheese
- "Meat loaf containing vegetables such as potatoes, onions, corn, carrots, and cabbage"

While each of these items satisfy at least one of the aspects in the query, only the third option satisfied both aspects in the query. Of the 500 queries in the RecipeMPR dataset, 375 (75%) of the queries include two aspects, 110 (22%) of the queries include three aspects, and the remaining 15 (3%) queries include four aspects. Additionally, each query is tagged as belonging to one or more of the following categories based on its attributes:

1. Specific: mentions a certain dish name (e.g., "spaghetti carbonara")
2. Analogical: uses metaphors or similes (e.g., "like McDonald's")
3. Commonsense: requires commonsense reasoning (e.g., "I'm watching my weight")
4. Temporal: explicitly references time of day or time (e.g., "slow", "fast")
5. Negated: includes a negation (e.g., "but", "without", "doesn't")

For example, the query "I would like a meat lasagna but I'm watching my weight" is tagged as being Specific, Commonsense, and Negated.

Zhang *et al.* [4] present results for a variety of naive algorithms on this dataset, including sparse retrieval, dense retrieval, and large language model prompting techniques. The best-performing approach was few-shot prompting with GPT-3 [16], yielding an accuracy of 83.4%. Zero-shot prompting with GPT-3 yielded an accuracy of 72.6%, and dense retrieval with GPT-3 embeddings yielded an accuracy of 54.0%. All other methods yielded an accuracy of 32% or lower.

They also consider aspect-based approaches for each of these algorithms, testing four different aggregation methods for aspect-based fusion. In their work, aspect-based approaches provide a small improvement in accuracy for some algorithms, but lower the performance for other algorithms. For example, aspect-based reasoning yielded an accuracy of 57.4% with few-shot prompting of GPT-3, 67.6% with zero-shot prompting of GPT-3, and 48.4% with dense retrieval using GPT-3 embeddings. The biggest improvement in performance came from using dense retrieval with TAS-B embeddings. This algorithm achieved an accuracy of 31.2% in the naive approach, but up to 36.4% for the aspect-based approaches.

4.2.2 Dataset 2: Multi-aspect RIRD

Note that although the RecipeMPR dataset includes multi-aspect queries by design, it does not include item reviews. Therefore, we introduce a second dataset which allows us to test our algorithms for reviewed-item retrieval.

The RIRD dataset compiles Yelp reviews for 50 different restaurants in Toronto and relevance assessments for these restaurants on a variety of queries. For our experiments, we manually select a subset of five queries which include the desired multi-aspect structure, and manually specify the aspects in these queries. These five queries are listed below:

- "Italian place with a burger"
- "A cafe that also offers beer"
- "Japanese restaurant with pasta"
- "An ice cream shop with bubble tea"
- "I am in search of a fancy Pakistani restaurant with authentic food"

Furthermore, while the RIRD dataset includes hundreds of review for each restaurant, we simply keep the first five reviews in the dataset for each restaurant. This greatly reduces the computational cost of implementing our entailment-based methods, since each review

must be evaluated against each query aspect.

As an example, the sole correct option for the query "Italian place with a burger" is a restaurant called "Gusto 101". Some examples of phrases in the reviews for this restaurant which indicate its suitability for this query include:

- " ... my boyfriend ordered the Gusto burger and he found it to be flavourful ... "
- " ... We all got pasta and I got the Mafalde ai Funghi ... "
- "... The burger was cooked to perfection and tasted so succulent ... "
- " ... I ordered the cacio e pepe pasta which was really good ... "

Note that only two of the five reviews for this restaurant mention "burger", while three of the five reviews for this restaurant mention Italian dishes such as pasta. Only one review mentions both of these aspects.

Abdollah Pour *et al.* [6] provide evaluation results for a variety of early fusion and late fusion algorithms applied to the RIRD dataset. However, since we use a small subset of the queries and reviews in the original RIRD dataset, our results are not directly comparable to their benchmarks.

4.3 Evaluation Metrics

We consider a variety of metrics to evaluate our implementation and compare the results against the benchmarks.

For the RecipeMPR dataset, each query includes one correct option and four incorrect options. Zhang *et al.* [4] therefore use *accuracy* to evaluate the performance of various algorithms on this dataset. We also compute the accuracy of our systems in order to compare our results to the benchmarks established in their work, and evaluate our algorithms' ability to correctly detect the item that fully satisfies each query.

For the multi-aspect RIRD dataset, there are fifty candidate items and only five queries. Since algorithms are much less likely to select the correct item from a pool of fifty candidates compared to five candidates in RecipeMPR, the accuracy metric is less useful for this dataset. Furthermore, the low sample size of queries means that accuracy can only take on one of six possible values, making it difficult to compare across methods. Therefore, we also evaluate the *mean reciprocal rank* (MRR) performance of our systems. MRR is computed as the

average of the reciprocals of the ranks assigned to the correct item in the ranking of items for a given query. A perfect MRR is a score of 1, and lower scores indicate inferior rankings.

One other limitation of using accuracy to evaluate the systems' performance is that ranking the correct item second would be treated the same as ranking the correct item last. We therefore also consider the *mean rank* assigned to the correct item to evaluate performance on RecipeMPR. For the multi-aspect RIRD dataset, the ranks assigned to the correct item tended to be left-skewed, with some higher values as outliers. Due to this skew, and the low sample size of queries for that dataset, we report the *median rank* assigned to the correct item instead of the mean rank. For both these metrics, a lower score indicates superior rankings.

Formally, let y_i be the ranking of the correct item for query i produced by a given algorithm. Then, accuracy across a dataset containing Q queries is calculated as:

$$\text{Accuracy} = \frac{1}{Q}\sum_{i=1}^{Q}\mathbb{I}[y_i = 1] \tag{4.1}$$

The mean reciprocal rank is calculated as:

$$\text{MRR} = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{y_i} \tag{4.2}$$

The mean rank is calculated as:

$$\text{Mean Rank} = \frac{1}{Q}\sum_{i=1}^{Q} y_i \tag{4.3}$$

To calculate the median rank, we assume the y_i are first sorted in ascending order and placed in an ordered list X, such that $X[1] = \min_i y_i$ and $X[Q] = \max_i y_i$. Then, the median rank is calculated as:

$$\text{Median Rank} = \begin{cases} X[\frac{n+1}{2}], & \text{if } Q \text{ is odd} \\ \frac{1}{2}\left(X[\frac{n}{2}] + X[\frac{n+1}{2}]\right), & \text{if } Q \text{ is even} \end{cases} \tag{4.4}$$

Chapter 5

Results

In this section, we present the results of our empirical evaluations and discuss the insights they give into our research questions. In addition to the plots presented below, our results are provided in tabular form in Appendix B.

5.1 RecipeMPR Results

RQ1. Are pre-trained textual entailment models competitive rankers for multi-aspect queries?

We present our results for the RecipeMPR dataset in Figure 5.1. From these results, it is clear that the overall performance on RecipeMPR when using the entailment model is much higher compared to using dense retrieval. This pattern holds for each of the three metrics we consider. In fact, the entailment approach achieves better accuracy than almost all the approaches presented in Zhang *et al.* [4], except for the large language model prompting methods with GPT-3. Notably, our entailment approach significantly improves upon the performance achieved using large language model prompting with GPT-2 [29], a model with 1.5 billion parameters, and using dense retrieval with embeddings from GPT-3 [16], which includes 175 billion parameters.

These results provide evidence that pre-trained textual entailment models can be competitive rankers for multi-aspect queries, at a lower computation cost than prompting large language models. This offers a promising direction for future research, such as integrating textual entailment models into multi-stage information retrieval pipelines.

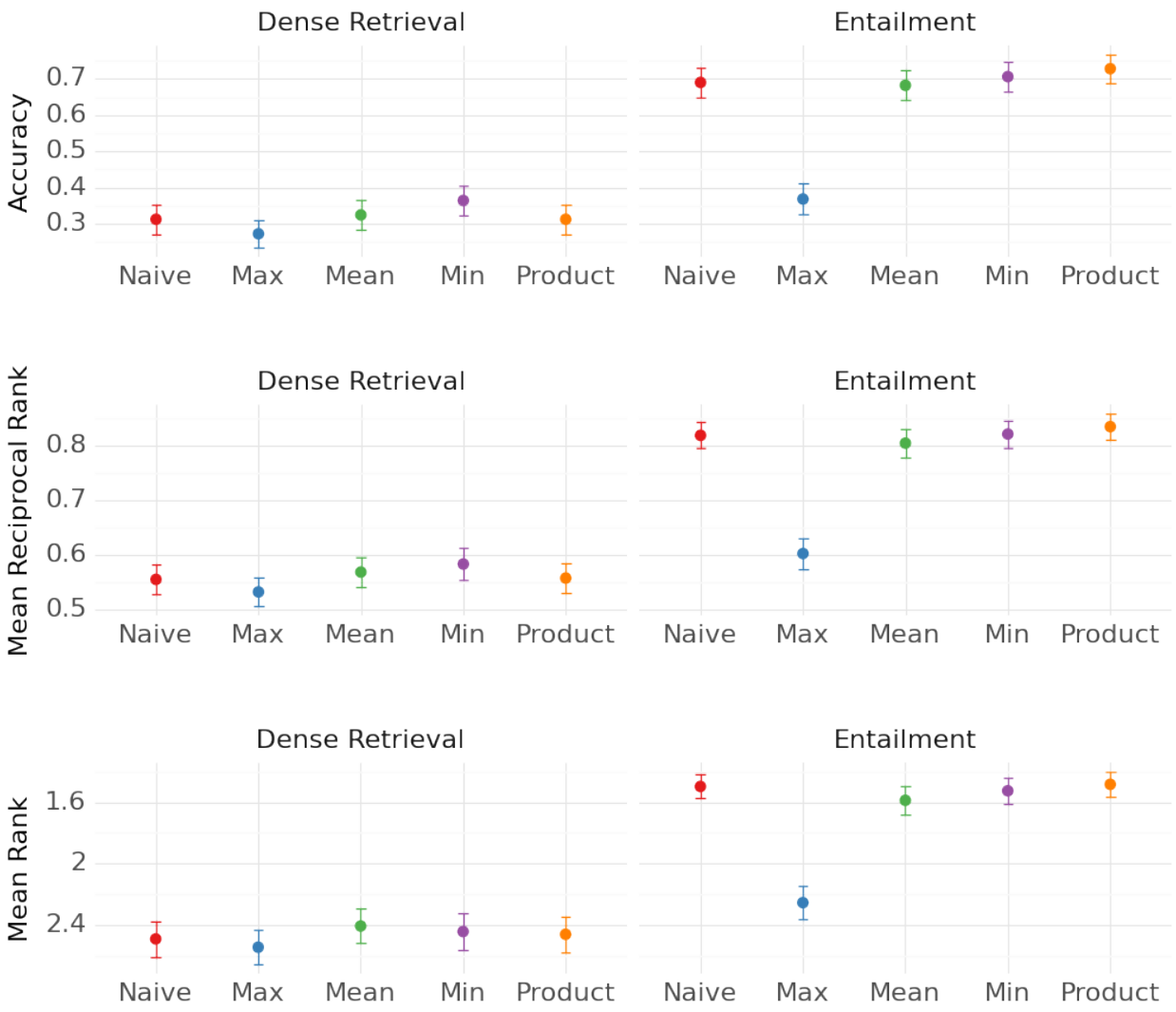

Figure 5.1: Comparison of results for RecipeMPR dataset. The error bars indicate 95% confidence intervals.

RQ2. Which aggregation methods yield the best performance for aspect-based fusion with multi-aspect queries?

From Figure 5.1, we see that for both dense retrieval and textual entailment, certain aggregation methods for aspect-based fusion offer an improvement in performance compared to the naive approach. For dense retrieval, the "Min", "Mean", and "Product" aggregation methods outperform the "Naive" baseline on all three metrics. For entailment, the "Product" aggregation method improves upon the "Naive" baseline on all three metrics. This method is the best-performing approach overall, and is also consistent with the objective presented in Equation 3.3, confirming that the principled approach based on our graphical model can

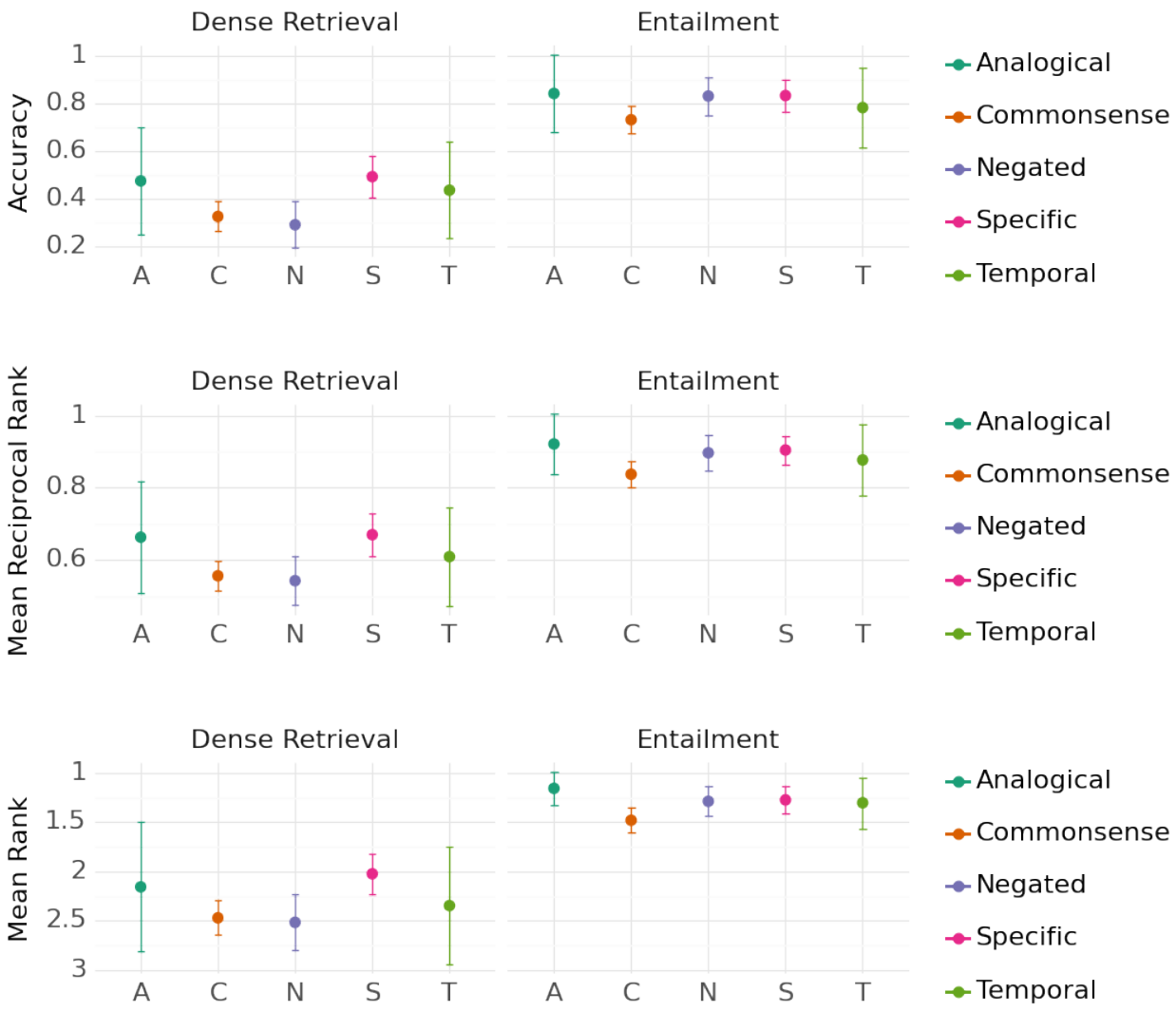

Figure 5.2: Comparison of results for best-performing method on each query type in the RecipeMPR dataset. The error bars indicate 95% confidence intervals.

be applied successfully in practice. Overall, "Product" aggregation seems to offer the best performance for aspect-based fusion, but "Mean" and "Min" aggregation should also be considered to identify the best approach for a given task.

For both dense retrieval and entailment, the "Max" aggregation method is consistently the worst-performing method. This result can be explained by the fact that negative examples in the RecipeMPR dataset are intentionally selected to satisfy at least one aspect in the query. This means that negative examples are relatively likely to achieve a high score on at least one of the aspects, and therefore likely to perform well on "Max" aggregation.

RQ3. How does the performance of retrieval algorithms vary for multi-aspect queries with particular attributes?

In Figure 5.2, we show the performance of the best-performing aggregation method for each query type, as described in 4.2. We see that there is are some significant variations in performance for different types of queries.

For example, we see that dense retrieval performs quite poorly for queries that include negation, whereas the entailment model performs reasonably well for these queries. Text containing negation is a well-known problem in neural information retrieval [30]. The superior performance when using an entailment model therefore offers an interesting direction for future work on improved handling of negation in information retrieval.

Furthermore, both architectures perform relatively poorly for queries that require common-sense reasoning. This result re-affirms findings from the literature that large language models lag significantly behind human-level performance on common-sense reasoning tasks [31]. Since common-sense reasoning can be particularly important to handle in conversational recommendation settings, the exploration of architectures that can effectively cope with such queries remains as an important area for future work.

In contrast, both the dense retrieval and textual entailment scoring functions perform quite well on analogical queries, suggesting that these models are able to encode the meaning of metaphors or similes in analogical queries. This result reaffirms the findings of Gao *et al.* [32] that standard neural architecture for natural language processing perform surprisingly well on metaphor detection tasks.

5.2 Multi-aspect RIRD Results

RQ4. Are pre-trained textual entailment models competitive scoring functions for naive late fusion with multi-aspect queries?

We present the results for the multi-aspect RIRD dataset in Figure 5.3. We see from these results that naive late fusion with a textual entailment model as the scoring function performs significantly worse than with a dense retrieval scoring function on this dataset. This is in contrast to the RecipeMPR results, where the entailment model significantly outperformed dense retrieval.

This finding suggests that the textual entailment model suffers more strongly from the failure mode described in Figure 3.4 than the dense retrieval model. Furthermore, it is

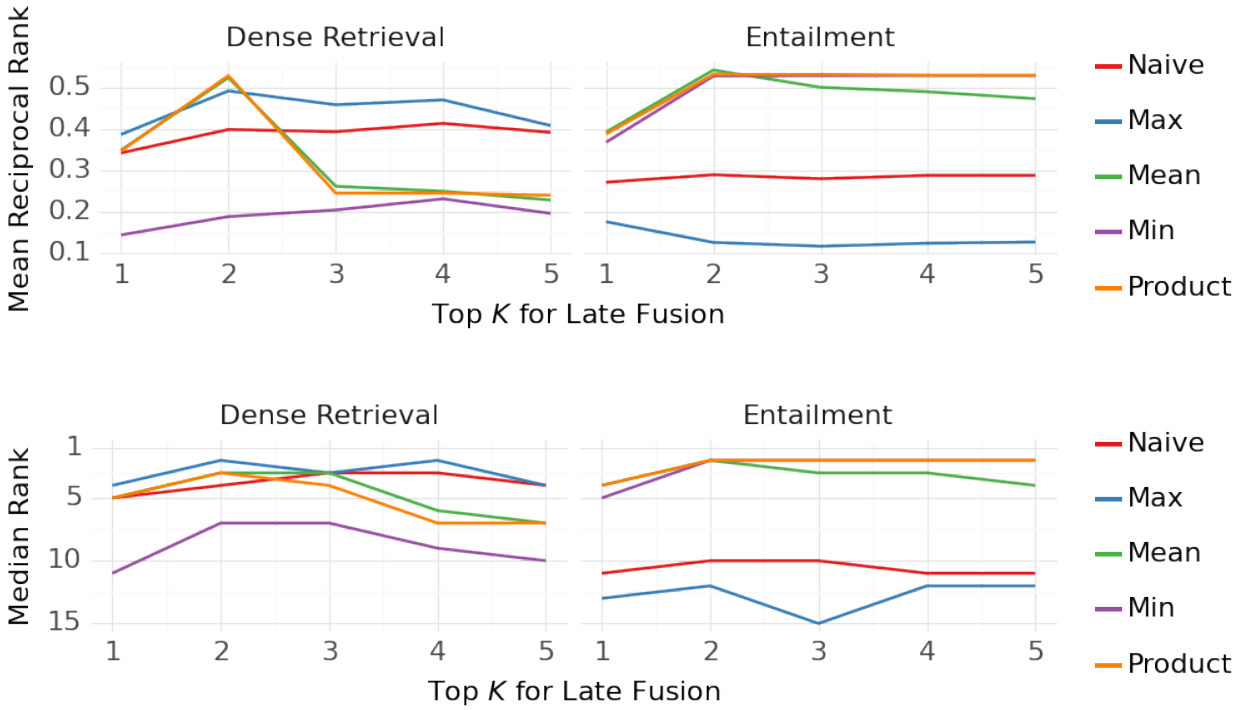

Figure 5.3: Comparison of results for multi-aspect RIRD dataset as a function of the parameter K used for late fusion.

possible that the different domains of the two datasets lead to differences in the models' performances. Another potential explanation for the gap in performance is that entailment models may not be well-suited to handle longer documents, such as reviews, which include text that is not directly related to the query.

RQ5. How do the choices of scoring function, aggregation method, and hyperparameter K impact the performance of aspect-based late fusion for multi-aspect reviewed-item retrieval?

We see in Figure 5.3 that for aspect-based late fusion with dense retrieval, the "Max" aggregation method consistently improves upon the naive late fusion benchmark for all settings of K. We also find that the "Mean" and "Product" aggregation methods improve upon naive late fusion for $K = 2$. When using the textual entailment scoring function, the "Mean", "Min", and "Product" aggregation methods each significantly improve upon the performance of the naive late fusion approach. These results suggest that aspect-based fusion indeed addresses the failure mode illustrated in Figure 3.4.

Interestingly, despite the fact that dense retrieval is clearly preferable to entailment for naive late fusion, the textual entailment scoring function actually yields better performance

for aspect-based late fusion. In particular, using an entailment scoring function with "Product" aggregation is the best performing method for all settings of K. This approach is consistent with the objective presented in Equation 3.3, confirming that a principled approach to estimate the probability of relevance for each item yields improved performance in practice. This result highlights the importance of selecting appropriate tools to combine with the principled algorithms we propose.

In our experiments, the best choice of the hyperparameter K for aspect-based late fusion appears to be $K = 2$. In general, the performance of competitive aspect-based late fusion methods tends to decrease slightly for higher values of K, except for the "Product" and "Min" aggregation methods with the entailment scoring function. Interestingly, late fusion with $K = 2$ consistently improves upon late fusion with $K = 1$, suggesting that in our dataset, information should be combined from two or more reviews to determine how well an item addresses a given aspect. It would be interesting to consider the effects of varying K for experiments with a larger set of reviews, such as the original RIRD dataset.

Lastly, the strong performance of "Max" aggregation with dense retrieval is in contrast to the results on RecipeMPR, where "Max" aggregation was consistently the worst-performing method. One possible explanation for this difference is that unlike the RecipeMPR dataset, the negative examples in the RIRD dataset were not intentionally designed to be hard negatives. Therefore, a restaurant that closely matches at least one of the aspects in a given query is quite likely to be relevant for that query. Despite this, "Max" remains the worst-performing aggregation method for the textual entailment model. This discrepancy for dense retrieval warrants further investigation.

RQ6. What are potential failure modes for aspect-based late fusion?

In Figure 5.4, we plot the ranks assigned to the correct item for each of the queries. Better performance is indicated by having a higher concentration of points at low ranks. We see that there are several outliers with unusually high ranks given to the correct item.

For the dense retrieval scoring function, the results are fairly consistent with the exception of two queries that yield poor performance with "Min" aggregation. These queries are "I am in search of a fancy Pakistani restaurant with authentic food" and "Japanese restaurant with pasta". For the first of these queries, the five reviews for the correct item do not give any evidence that the restaurant satisfies the aspect "fancy", which resulted in fairly low scores for this aspect. Similarly, for the second of these queries, none of the reviews for the

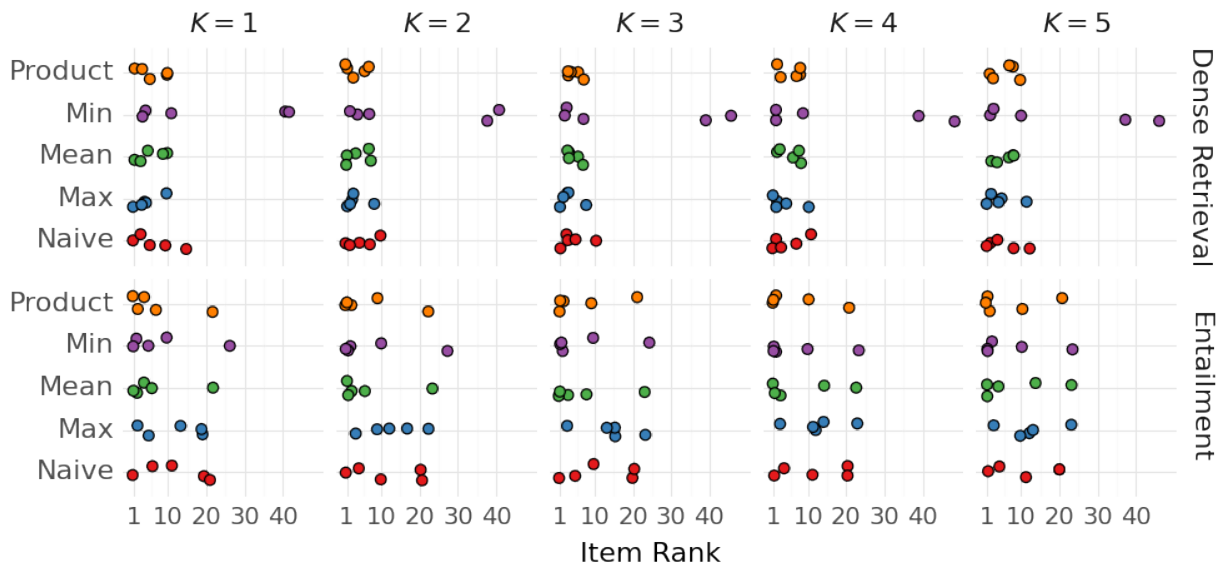

Figure 5.4: Ranks assigned to correct items in multi-aspect RIRD

correct item actually describe pasta. For the entailment scoring function, the query "An ice cream shop with bubble tea" consistently has the worst results. Since none of the reviews of the correct item specifically mention "bubble tea", it is not surprising that this query poses a challenge.

In summary, we reveal a failure mode for aspect-based late fusion in cases when none of the reviews for the correct item provide evidence of satisfying a given aspect. The score of the correct item is likely to be low in these cases, harming the resulting ranking. Expanding our dataset to include a bigger number of reviews or queries with multiple correct options could alleviate the effect of this issue on our results. One other possible way to mitigate this issue is by introducing a weighting over aspects in the query to downweight aspects which fail to clearly differentiate items. For instance, if all candidate items score poorly on a given aspect, that aspect could be downweighted in the aggregation.

Chapter 6

Conclusion

We conclude by summarizing the main contributions of this thesis, identifying a few key limitations of our work, and discussing directions for future research based on our findings.

6.1 Summary of Contributions

In this thesis, we develop graphical models to represent the multi-aspect retrieval, reviewed-item retrieval, and multi-aspect reviewed-item retrieval settings. These graphical models shed light on the theoretical foundations of previously-proposed methods. In particular, we argue that multi-aspect queries are a potential failure mode for pre-existing reviewed-item retrieval algorithms. Based on our graphical models, we derive principled algorithms to account for multi-aspect queries. Notably, we propose an algorithm that addresses this multi-aspect failure mode for reviewed-item retrieval.

We show empirically that these principled algorithms offer improved performance on multi-aspect queries, including in the reviewed-item retrieval setting. We also show that using pre-trained large language entailment models, a novel approach to information retrieval, can achieve superior performance to dense retrieval. These entailment models lend themselves particularly well to the probabilistic interpretations prescribed by our principled algorithms.

6.2 Limitations

One limitation of our experimental setup is that the RecipeMPR dataset we used for evaluation was designed with specific constraints that may not reflect all real-world queries. For example, not all multi-aspect queries might conform with our assumption that a relevant item must satisfy all aspects in the query. It is worth exploring the performance of aspect-based approaches for multi-aspect retrieval on additional datasets whose structure may not be as rigid as RecipeMPR.

Additionally, in all our experiments, we assumed that the query aspects were known. However, in realistic settings, aspects would need to be automatically extracted from queries. Evaluating the impact of automatic query extraction on the performance of aspect-based methods would be an important next step before implementing an aspect-based pipeline in practice.

Finally, we presented an experimental evaluation of aspect-based late fusion on a fairly small number of queries and reviews. Evaluating this algorithm on a larger dataset, including more queries and more reviews per item, would give a more useful indication about the algorithm's merits in practice.

6.3 Directions for Future Work

Our work offers several promising directions for future research.

Firstly, although we show that aspect-based approaches can improve performance for multi-aspect retrieval, a better understanding of query characteristics that make them well-suited for specific aspect-based approaches would be valuable. For instance, this understanding could help design systems that adaptively select for each individual query the most appropriate method, or even determine the ideal weights to ensemble predictions from multiple methods.

Secondly, our graphical model framework for reviewed-item retrieval motivates a Bayesian marginalization approach to early fusion that has not yet been explored. Previously-proposed early fusion algorithms can be interpreted as assuming that review embeddings are normally-distributed about a latent item embedding. However, since different item reviews often describe distinct aspects of an item, a Gaussian mixture model may be a more appropriate assumption. Marginalizing over the posterior for the latent item representation under various assumptions for the distribution of review embeddings offers an interesting direction for

further research on reviewed-item retrieval.

Lastly, we argue that algorithms based on theoretical foundations, combined with appropriate tools, can yield performances that are competitive with prompting large language models at a lower computational cost. Particularly, we use a pre-trained textual entailment model to obtain probabilistic scores that can be combined in a principled way for multi-aspect retrieval. Our successful application of pre-trained entailment models motivates further investigation into their use in information retrieval, and especially into including such models within multi-stage information retrieval pipelines.

Bibliography

[1] C. K. Reddy, L. Màrquez, F. Valero, *et al.*, "Shopping queries dataset: A large-scale ESCI benchmark for improving product search," 2022. arXiv: 2206.06588.

[2] J. Gao, C. Xiong, P. Bennett, and N. Craswell, *Neural Approaches to Conversational Information Retrieval* (The Information Retrieval Series). Springer, 2023, vol. 44, ISBN: 978-3-031-23079-0. DOI: 10.1007/978-3-031-23080-6. [Online]. Available: https://doi.org/10.1007/978-3-031-23080-6.

[3] W. Kong, S. Khadanga, C. Li, *et al.*, "Multi-aspect dense retrieval," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '22, Washington DC, USA: Association for Computing Machinery, 2022, pp. 3178–3186, ISBN: 9781450393850. DOI: 10.1145/3534678.3539137. [Online]. Available: https://doi.org/10.1145/3534678.3539137.

[4] H. Zhang, A. Korikov, P. Farinneya, *et al.*, "Recipe-MPR: A test collection for evaluating multi-aspect preference-based natural language retrieval," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '23, Taipei, Taiwan: Association for Computing Machinery, 2023, pp. 2744–2753, ISBN: 9781450394086. DOI: 10.1145/3539618.3591880. [Online]. Available: https://doi.org/10.1145/3539618.3591880.

[5] N. Asghar, "Yelp dataset challenge: Review rating prediction," *ArXiv*, vol. abs/1605.05362, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:18058702.

[6] M. M. Abdollah Pour, P. Farinneya, A. Toroghi, *et al.*, "Self-supervised contrastive BERT fine-tuning for fusion-based reviewed-item retrieval," in *Advances in Information Retrieval: 45th European Conference on Information Retrieval*, Dublin, Ireland: Springer-Verlag, Apr. 2023, pp. 3–17, ISBN: 978-3-031-28243-0. DOI: 10.1007/978-3-031-28244-7_1. [Online]. Available: https://doi.org/10.1007/978-3-031-28244-7%5C%5F1.

[7] S. Zhang and K. Balog, "Design patterns for fusion-based object retrieval," in *Advances in Information Retrieval - 39th European Conference on IR Research, ECIR 2017,*

Aberdeen, UK, April 8-13, 2017, Proceedings, J. M. Jose, C. Hauff, I. S. Altingövde, *et al.*, Eds., ser. Lecture Notes in Computer Science, vol. 10193, 2017, pp. 684–690. DOI: 10.1007/978-3-319-56608-5_66. [Online]. Available: https://doi.org/10.1007/978-3-319-56608-5%5C%5F66.

[8] I. Titov and R. McDonald, "A joint model of text and aspect ratings for sentiment summarization," in *Proceedings of ACL-08: HLT*, J. D. Moore, S. Teufel, J. Allan, and S. Furui, Eds., Columbus, Ohio: Association for Computational Linguistics, Jun. 2008, pp. 308–316. [Online]. Available: https://aclanthology.org/P08-1036.

[9] S. Robertson, "The probability ranking principle in IR," *Journal of Documentation*, vol. 33, no. 4, pp. 294–304, 1977. DOI: 10.1108/eb026647. [Online]. Available: https://doi.org/10.1108/eb026647.

[10] S. Sanner, S. Guo, T. Graepel, S. Kharazmi, and S. Karimi, "Diverse retrieval via greedy optimization of expected 1-call@k in a latent subtopic relevance model," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, ser. CIKM '11, Glasgow, Scotland, UK: Association for Computing Machinery, 2011, pp. 1977–1980, ISBN: 9781450307178. DOI: 10.1145/2063576.2063869. [Online]. Available: https://doi.org/10.1145/2063576.2063869.

[11] M. Sanderson, "Test collection based evaluation of information retrieval systems," *Foundations and Trends in Information Retrieval*, vol. 4, no. 4, pp. 247–375, 2010, ISSN: 1554-0669. DOI: 10.1561/1500000009. [Online]. Available: http://dx.doi.org/10.1561/1500000009.

[12] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Eng. Bull.*, vol. 24, no. 4, pp. 35–43, 2001. [Online]. Available: http://sites.computer.org/debull/A01DEC-CD.pdf.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *North American Chapter of the Association for Computational Linguistics*, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:52967399.

[14] V. Karpukhin, B. Oguz, S. Min, *et al.*, "Dense passage retrieval for open-domain question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 6769–6781. DOI: 10.18653/v1/2020.emnlp-main.550. [Online]. Available: https://aclanthology.org/2020.emnlp-main.550.

[15] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, *et*

al., Eds., vol. 30, Curran Associates, Inc., 2017. DOI: 10.5555/3295222.3295349. [Online]. Available: https://proceedings.neurips.cc/paper%5C%5Ffiles/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[16] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper%5C%5Ffiles/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[17] R. Nogueira, Z. Jiang, R. Pradeep, and J. Lin, "Document ranking with a pretrained sequence-to-sequence model," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds., Online: Association for Computational Linguistics, Nov. 2020, pp. 708–718. DOI: 10.18653/v1/2020.findings-emnlp.63. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.63.

[18] C. Raffel, N. Shazeer, A. Roberts, *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html.

[19] X. Ma, X. Zhang, R. Pradeep, and J. Lin, "Zero-shot listwise document reranking with a large language model," *CoRR*, vol. abs/2305.02156, 2023. DOI: 10.48550/arxiv.2305.02156. arXiv: 2305.02156. [Online]. Available: https://doi.org/10.48550/arXiv.2305.02156.

[20] Y. Zhu, H. Yuan, S. Wang, *et al.*, "Large language models for information retrieval: A survey," *CoRR*, vol. abs/2308.07107, 2023. arXiv: 2306.07401. [Online]. Available: https://arxiv.org/abs/2308.07107.

[21] S. Clinchant, C. Goutte, and E. Gaussier, "Lexical entailment for information retrieval," in *Advances in Information Retrieval*, M. Lalmas, A. MacFarlane, S. Rüger, A. Tombros, T. Tsikrika, and A. Yavlinsky, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 217–228, ISBN: 978-3-540-33348-7. [Online]. Available: https://link.springer.com/chapter/10.1007/11735106%5C%5F20.

[22] C. L. Clarke, M. Kolla, G. V. Cormack, *et al.*, "Novelty and diversity in information retrieval evaluation," in *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '08, Singapore, Singapore: Association for Computing Machinery, 2008, pp. 659–666, ISBN: 9781605581644. DOI: 10.1145/1390334.1390446. [Online]. Available: https://doi.org/10.1145/1390334.1390446.

[23] V. Bursztyn, J. Healey, N. Lipka, E. Koh, D. Downey, and L. Birnbaum, "It doesn't look good for a date: Transforming critiques into preferences for conversational recommendation systems," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds., Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 1913–1918. DOI: 10.18653/v1/2021.emnlp-main.145. [Online]. Available: https://aclanthology.org/2021.emnlp-main.145.

[24] M. Lewis, Y. Liu, N. Goyal, *et al.*, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds., Online: Association for Computational Linguistics, Jul. 2020, pp. 7871–7880. DOI: 10.18653/v1/2020.acl-main.703. [Online]. Available: https://aclanthology.org/2020.acl-main.703.

[25] A. Williams, N. Nangia, and S. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, Louisiana: Association for Computational Linguistics, 2018, pp. 1112–1122. [Online]. Available: http://aclweb.org/anthology/N18-1101.

[26] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," *ArXiv*, vol. abs/1910.01108, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:203626972.

[27] T. Nguyen, M. Rosenberg, X. Song, *et al.*, "MS MARCO: A human generated machine reading comprehension dataset," in *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, T. R. Besold, A. Bordes, A. S. d'Avila Garcez, and G. Wayne, Eds., ser. CEUR Workshop Proceedings, vol. 1773, CEUR-WS.org, 2016. [Online]. Available: https://ceur-ws.org/Vol-1773/CoCoNIPS%5C%5F2016%5C%5Fpaper9.pdf.

[28] S. Hofstätter, S.-C. Lin, J.-H. Yang, J. Lin, and A. Hanbury, "Efficiently teaching an effective dense retriever with balanced topic aware sampling," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '21, Virtual Event: Association for Computing Machinery, 2021, pp. 113–122, ISBN: 9781450380379. DOI: 10.1145/3404835.3462891. [Online]. Available: https://doi.org/10.1145/3404835.3462891.

[29] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019. [Online]. Available: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

[30] O. Weller, D. Lawrie, and B. Van Durme, "NevIR: Negation in neural information retrieval," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, Y. Graham and M. Purver, Eds., St. Julian's, Malta: Association for Computational Linguistics, Mar. 2024, pp. 2274–2287. [Online]. Available: https://aclanthology.org/2024.eacl-long.139.

[31] X. L. Li, A. Kuncoro, J. Hoffmann, C. de Masson d'Autume, P. Blunsom, and A. Nematzadeh, "A systematic investigation of commonsense knowledge in large language models," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds., Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 11 838–11 855. DOI: 10.18653/v1/2022.emnlp-main.812. [Online]. Available: https://aclanthology.org/2022.emnlp-main.812.

[32] G. Gao, E. Choi, Y. Choi, and L. Zettlemoyer, "Neural metaphor detection in context," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds., Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 607–613. DOI: 10.18653/v1/D18-1060. [Online]. Available: https://aclanthology.org/D18-1060.

Appendix A

Theoretical Justification for Aspect-based Late Fusion

In this appendix, we provide a theoretical argument for using aspect-based late fusion instead of naive late fusion for multi-aspect queries. Specifically, we show that under certain assumptions, aspect-based late fusion and naive late fusion both offer unbiased estimates of the true probability of relevance, but aspect-based late fusion offers a lower-variance estimate.

A.1 Setup

We consider a query q with two aspects, q_1 and q_2. We focus on this two-aspect case, though we conjecture that our proof extends to the case of three or more aspects. We wish to estimate the probability of relevance $P(r_i|q, s_i)$ of item s_i from a set of $m \geq 2$ reviews d_{ik}.

We define $a = P(c_{i1}|s_i, q_1)$ and $b = P(c_{i2}|s_i, q_2)$. According to the graphical model in Figure 3.3, the true value which we seek to estimate is:

$$\theta = P(r_i|q, s_i) = P(c_{i1}|s_i, q_1)P(c_{i2}|s_i, q_2) = ab \tag{A.1}$$

A.2 Assumptions

Our proof relies on the following assumptions.

Assumption 1. $P(c_{ij}|s_i, q_j) \in [0, 1]$ *for all* j

This assumption implies that we can never be fully certain whether an item satisfies a given query aspect. This is not a very restrictive assumption since the aspect coverage probabilities can still be made arbitrarily close to 0 or 1. However, it allows us to assume that $0 < a, b < 1$, which will be useful for the proof later on.

Assumption 2. *Let g be a scoring function for aspect-based late fusion. Then, for each aspect q_j, the scores $g(d_{ik}, q_j)$ are independent and identically-distributed.*

This assumption is similar to the setup of the graphical model in Figure 3.3. It is reasonable to assume that since the reviews are independent and identically distributed for a given aspect, their scores should also be independent and identically distributed.

Assumption 3. *Let g be a scoring function for aspect-based late fusion. Then, $g(d_{ik}, q_j)$ are Bernoulli random variables with probabilities $P(c_{ij}|s_i, q_j)$.*

This assumption contends that our scoring function g gives us an unbiased binary estimate of the true aspect coverage probability. This assumption is somewhat restrictive, but with recent advancements in natural language understanding via large language models, not completely far-fetched. In practice, we observe that using a pre-trained entailment model as the scoring function already tends to give estimates close to 0 or 1 in many cases.

Assumption 4. *Let f be the scoring function for naive late fusion and g be the scoring function for aspect-based late fusion. Then, $f(d_{ik}, q) = \prod_j g(d_{ik}, q_j)$.*

This assumption essentially says that the scoring function used for naive late fusion correctly accounts for the multi-aspect structure of the query, making it in some sense optimal. Our main result is therefore quite strong, as it shows that aspect-based late fusion improves upon naive late fusion even when we use an optimal scoring function for naive late fusion.

A.3 Proof

Our proof is structured as follows. In Theorem 1, we show that naive late fusion gives an unbiased estimates of the true probability of relevance θ. In Theorem 2, we show that aspect-based late fusion also gives an unbiased estimate of θ. Lastly, in Theorem 3, we show that aspect-based late fusion gives an estimator that has lower variance than the estimator given by naive late fusion.

Theorem 1. *Under Assumptions 2, 3 and 4, naive late fusion gives an unbiased estimate of the true probability of relevance θ.*

Proof. We wish to show that $\mathbb{E}[\hat{\theta}_{\text{NLF}}] = \theta$. Our proof proceeds from the definition of naive late fusion as follows:

$$\begin{aligned}
\mathbb{E}\left[\hat{\theta}_{\text{NLF}}\right] &= \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} f(d_{ik}, q)\right] && \text{from Equation 3.5} \\
&= \frac{1}{m}\sum_{k=1}^{m} \mathbb{E}\left[g(d_{ik}, q_1)g(d_{ik}, q_2)\right] && \text{from Assumption 4} \\
&= \frac{1}{m}\sum_{k=1}^{m} \mathbb{E}[g(d_{ik}, q_1)]\,\mathbb{E}[g(d_{ik}, q_2)] && \text{from Assumption 2} \\
&= \frac{1}{m}\sum_{k=1}^{m} ab && \text{from Assumption 3} \\
&= ab \\
&= \theta && \square
\end{aligned}$$

Theorem 2. *Under Assumptions 2, 3 and 4, aspect-based late fusion gives an unbiased estimate of the true probability of relevance θ.*

Proof. We wish to show that $\mathbb{E}[\hat{\theta}_{\text{ABLF}}] = \theta$. Our proof proceeds from the definition of aspect-based late fusion as follows:

$$\begin{aligned}
\mathbb{E}\left[\hat{\theta}_{\text{ABLF}}\right] &= \mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_1)\right)\left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_2)\right)\right] && \text{from Equation 3.6} \\
&= \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_1)\right]\mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_2)\right] && \text{from Assumption 2} \\
&= \left(\frac{1}{m}\sum_{k=1}^{m} \mathbb{E}[g(d_{ik}, q_1)]\right)\left(\frac{1}{m}\sum_{k=1}^{m} \mathbb{E}[g(d_{ik}, q_2)]\right) \\
&= \left(\frac{1}{m}\sum_{k=1}^{m} a\right)\left(\frac{1}{m}\sum_{k=1}^{m} b\right) && \text{from Assumption 3} \\
&= ab \\
&= \theta && \square
\end{aligned}$$

Having proved that both naive late fusion and aspect-based late fusion provide unbiased estimators for the probability of relevance, we now show that aspect-based late fusion yields a lower-variance estimator, and is therefore more desirable.

Theorem 3. *Under Assumptions 4, 2 and 3, aspect-based late fusion gives a lower-variance estimator of the true probability of relevance θ than naive late fusion.*

In order to prove this theorem, we first prove three intermediate lemmas.

Lemma 3.1. *For independent binary random variables x_i with means p,*

$$\mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} x_i\right)^2\right] = \frac{p}{m}(1+(m-1)p)$$

Proof.

$$\begin{aligned}
\mathbb{E}\left[\left(\frac{1}{m}\sum_{i=1}^{m} x_i\right)^2\right] &= \frac{1}{m^2}\,\mathbb{E}\left[\sum_{i=1}^{m} x_i^2 + 2\sum_{i=1}^{m}\sum_{j=1}^{i-1} x_i x_j\right] \\
&= \frac{1}{m^2}\left(\sum_{i=1}^{m}\mathbb{E}[x_i^2] + 2\sum_{i=1}^{m}\sum_{j=1}^{i-1}\mathbb{E}\left[x_i x_j\right]\right) \\
&= \frac{1}{m^2}\left(\sum_{i=1}^{m}\mathbb{E}[x_i] + 2\sum_{i=1}^{m}\sum_{j=1}^{i-1}\mathbb{E}[x_i]\mathbb{E}[x_j]\right) \\
&= \frac{1}{m^2}\left(mp + 2\binom{m}{2}p^2\right) \\
&= \frac{p}{m}(1+(m-1)p)
\end{aligned}$$

□

Lemma 3.2. *The expectation for the mean of the square of the naive late fusion estimator is:*

$$\mathbb{E}[(\hat{\theta}_{NLF})^2] = \frac{ab}{m}\left(1+(m-1)ab\right)$$

Proof.

$$\begin{aligned}
\mathbb{E}[(\hat{\theta}_{\mathrm{NLF}})^2] &= \mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} f(d_{ik}, q)\right)^2\right] && \text{from Equation 3.5} \\
&= \frac{ab}{m}(1+(m-1)ab) && \text{by Lemma 3.1 with } p = ab
\end{aligned}$$

□

Lemma 3.3. *The expectation for the mean of the square of the aspect-based late fusion estimator is:*

$$\mathbb{E}[(\hat{\theta}_{ABLF})^2] = \frac{ab}{m^2}(1 + (m-1)(a+b) + (m-1)^2 a^2 b^2)$$

Proof.

$$\begin{aligned}
\mathbb{E}[(\hat{\theta}_{\text{ABLF}})^2] &= \mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_1)\right)^2 \left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_2)\right)^2\right] \\
&= \mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_1)\right)^2\right] \mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_2)\right)^2\right] && \text{from Assumption 2} \\
&= \frac{a}{m}(1 + (m-1)a)\, \mathbb{E}\left[\left(\frac{1}{m}\sum_{k=1}^{m} g(d_{ik}, q_2)\right)^2\right] && \text{by Lemma 3.1 with } p = a \\
&= \frac{a}{m}(1 + (m-1)a)\frac{b}{m}(1 + (m-1)b) && \text{by Lemma 3.1 with } p = b \\
&= \frac{ab}{m^2}(1 + (m-1)(a+b) + (m-1)^2 a^2 b^2)
\end{aligned}$$

□

Finally, we can prove Theorem 3.

Proof. We wish to show that $\text{Var}[\hat{\theta}_{\text{NLF}}] > \text{Var}[\hat{\theta}_{\text{ABLF}}]$. We do this with the following sequence of equivalent statements.

$$\begin{aligned}
\text{Var}[\hat{\theta}_{\text{NLF}}] &> \text{Var}[\hat{\theta}_{\text{ABLF}}] \\
\mathbb{E}[(\hat{\theta}_{\text{NLF}})^2] - \mathbb{E}[\hat{\theta}_{\text{NLF}}]^2 &> \mathbb{E}[(\hat{\theta}_{\text{ABLF}})^2] - \mathbb{E}[\hat{\theta}_{\text{ABLF}}]^2 \\
\mathbb{E}[(\hat{\theta}_{\text{NLF}})^2] - (ab)^2 &> \mathbb{E}[(\hat{\theta}_{\text{ABLF}})^2] - (ab)^2 && \text{from Theorems 1 and 2} \\
\mathbb{E}[(\hat{\theta}_{\text{NLF}})^2] &> \mathbb{E}[(\hat{\theta}_{\text{ABLF}})^2]
\end{aligned}$$

We proceed by using the results of Lemmas 3.2 and 3.3:

$$\begin{aligned}
\frac{ab}{m}\,(1+(m-1)ab) &> \frac{ab}{m^2}(1+(m-1)(a+b)+(m-1)^2a^2b^2) \\
m+m(m-1)ab &> 1+(m-1)(a+b)+(m-1)^2a^2b^2 && \text{multiply by } \tfrac{m^2}{ab}>0 \\
(m-1)+m(m-1)ab &> (m-1)(a+b)+(m-1)^2a^2b^2 \\
1+mab &> a+b+(m-1)a^2b^2 && \text{divide by } m-1>0 \\
1+ab(m-(m-1)ab) &> a+b && \text{(A.2)}
\end{aligned}$$

Since $0 < ab < 1$, we know $(m-1)ab < m-1$. Multiplying by -1, we have $-(m-1)ab > 1-m$. Adding m to both sides, we have $m-(m-1)ab > 1$. The left hand side of inequality A.2 is therefore greater than $1+ab$, so it suffices to show that:

$$\begin{aligned}
1+ab &> a+b \\
1-b &> a-ab \\
1-b &> a(1-b) \\
1 &> a && \text{since } 1-b>0 \text{ by Assumption 1}
\end{aligned}$$

Since by Assumption 1, a is indeed strictly less than 1, this concludes our proof. □

Appendix B

Tables of Results

B.1 RecipeMPR Results

Table B.1: RecipeMPR Results with 95% confidence intervals

Scorer	Method	MRR	Accuracy	Mean Rank
TAS-B	Naive	0.55 ± 0.03	0.31 ± 0.04	2.5 ± 0.1
	Max	0.53 ± 0.03	0.27 ± 0.04	2.5 ± 0.1
	Mean	0.57 ± 0.03	0.32 ± 0.04	2.4 ± 0.1
	Min	0.58 ± 0.03	0.36 ± 0.04	2.4 ± 0.1
	Product	0.56 ± 0.03	0.31 ± 0.04	2.5 ± 0.1
Entailment	Naive	0.82 ± 0.02	0.69 ± 0.04	1.5 ± 0.1
	Max	0.60 ± 0.03	0.37 ± 0.04	2.3 ± 0.1
	Mean	0.81 ± 0.03	0.68 ± 0.04	1.6 ± 0.1
	Min	0.82 ± 0.02	0.71 ± 0.04	1.5 ± 0.1
	Product	0.84 ± 0.02	0.73 ± 0.04	1.5 ± 0.1

B.2 Multi-aspect RIRD Results

Table B.2: Multi-aspect RIRD MMR results with 95% confidence intervals estimated via bootstrapping

Scorer	Method	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
TAS-B	Naive	0.34 ± 0.30	0.40 ± 0.29	0.39 ± 0.28	0.41 ± 0.29	0.39 ± 0.30
	Max	0.39 ± 0.28	0.49 ± 0.25	0.46 ± 0.26	0.47 ± 0.27	0.41 ± 0.28
	Mean	0.35 ± 0.29	0.52 ± 0.35	0.26 ± 0.08	0.25 ± 0.13	0.23 ± 0.13
	Min	0.14 ± 0.11	0.19 ± 0.15	0.20 ± 0.16	0.23 ± 0.19	0.20 ± 0.17
	Product	0.35 ± 0.30	0.53 ± 0.34	0.25 ± 0.07	0.25 ± 0.13	0.24 ± 0.13
Entailment	Naive	0.27 ± 0.32	0.29 ± 0.32	0.28 ± 0.32	0.29 ± 0.32	0.29 ± 0.32
	Max	0.18 ± 0.15	0.13 ± 0.09	0.12 ± 0.10	0.12 ± 0.09	0.13 ± 0.09
	Mean	0.39 ± 0.30	0.54 ± 0.35	0.50 ± 0.37	0.49 ± 0.38	0.47 ± 0.38
	Min	0.37 ± 0.31	0.53 ± 0.37	0.53 ± 0.37	0.53 ± 0.36	0.53 ± 0.36
	Product	0.39 ± 0.30	0.53 ± 0.36	0.53 ± 0.36	0.53 ± 0.36	0.53 ± 0.36

Table B.3: Multi-aspect RIRD median rank results with 95% confidence intervals estimated via bootstrapping

Scorer	Method	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
TAS-B	Naive	5 ± 6	4 ± 4	3 ± 3	3 ± 5	4 ± 5
	Max	4 ± 3	2 ± 2	3 ± 2	2 ± 3	4 ± 4
	Mean	5 ± 5	3 ± 4	3 ± 2	6 ± 4	7 ± 3
	Min	11 ± 31	7 ± 30	7 ± 32	9 ± 32	10 ± 29
	Product	5 ± 5	3 ± 4	4 ± 2	7 ± 4	7 ± 4
Entailment	Naive	11 ± 11	10 ± 13	10 ± 12	11 ± 12	11 ± 12
	Max	13 ± 11	12 ± 8	15 ± 6	12 ± 7	12 ± 7
	Mean	4 ± 8	2 ± 10	3 ± 10	3 ± 13	4 ± 12
	Min	5 ± 11	2 ± 12	2 ± 11	2 ± 11	2 ± 11
	Product	4 ± 9	2 ± 10	2 ± 10	2 ± 10	2 ± 10