Instance Attribution in Information Retrieval

Identifying and Selecting Influential Instances with Instance Attribution for Passage Re-Ranking

Ilgin Sara Hacipoglu

Delft University of Technology

TU Delft

Instance Attribution in Information Retrieval

Identifying and Selecting Influential Instances with Instance Attribution for Passage Re-Ranking

by

Student Name	Student Number
Ilgin Sara Hacipoglu	5569206

Instructor:	A. Anand
Daily Supervisor:	M. Idahl
Project Duration:	December, 2022 - October, 2023
Faculty:	Faculty of Electrical Engineering, Mathematics and Computer Science, Delft

TUDelft

Abstract

The complexity of deep neural rankers and large datasets make it increasingly more challenging to understand why a document is predicted as relevant to a given query. A growing body of work focuses on interpreting ranking models with different explainable AI methods. Instance attribution methods aim to explain individual predictions of machine learning models by identifying influential training data. However, despite their popularity, instance attribution methods are largely unexplored in the information retrieval context, particularly in text ranking. This thesis introduces an application of TracInCP, an instance attribution method, to infer the influence of query-passage training data on ranking model predictions. We propose and evaluate training data subset selection approaches based on influence. By analyzing patterns in influential examples, we find common query and passage characteristics in the training data that affect the model's ranking decisions. Finally, we demonstrate possible challenges in using instance attribution to create smaller datasets for text ranking tasks.

Preface

I would like to express my deepest gratitude to my supervisor, Prof. Avishek Anand, for his unwavering guidance and continuous support throughout my thesis journey. Initially, I was confused and worried that this work might exceed my skill level. Explainable AI was entirely foreign to me, a distant concept I could not wrap my head around. Thanks to our meetings, much of which were patient explanations and backtracks to the basics when I was confused, I gradually gained the confidence to explore this fascinating research area. With his invaluable insights and feedback, I have learned how to approach a research problem - not with shortcuts, however tempting, but the proper way.

I would like to thank Prof. Sicco Verwer for being on my thesis committee.

Max Idahl, thank you for being the greatest daily supervisor I could have ever wished for. Your patience in answering my ceaseless questions, continuous help, and, most importantly, steadfast company throughout this journey have been an absolute blessing. I began this thesis with you as my mentor, I now conclude it with the realization that this academic journey has also gifted me a cherished friend.

Inés and Sneha thank you for always being there, offering reassurance every time I doubted my abilities. Our meetups in B28 were what kept me motivated and excited to work on my thesis. You encouraged me to think creatively and, most importantly, inspired me. Andreas, I want to extend my heartfelt appreciation for patiently listening to my endless discussions about my thesis, and for tirelessly brainstorming with me when I felt stuck, even when you had no prior knowledge of the subject. Thank you for being my faithful cheerleader throughout this journey.

Finally, my parents, thank you for always being there, even when I am far away from home. I can't even begin to fathom being where I am today without your endless support and trust. I am deeply grateful that you encouraged me to broaden my horizons.

Ilgin Sara Hacipoglu
Delft, October 2023

Contents

List of Figures

List of Tables

1 Introduction

Over the last decade, Machine Learning has surged in various industries, bringing about a new era of technological advancements and products. We now routinely encounter Artificial Intelligence-driven technologies in our daily lives, such as digital assistants, smart homes, self-driving cars, and chatbots. As of 2021, a significant 76% of enterprises have already made AI their top priority in their budgets and future strategies [70]. The appeal of machine learning is its potential to learn complex patterns from data without hard-coded logic or pre-defined rules. Nevertheless, this flexibility also brings opacity. In critical applications today, scientists and researchers often steer clear of the state-of-the-art methods and the most robust models due to the inherent complexity and lack of explainability. Today, there is a prevailing perception of a tradeoff: while a powerful model can offer substantial business benefits, its lack of transparency introduces risk when deploying it in critical applications. Decisions made by AI systems can significantly impact some applications, such as telehealth or autonomous vehicles, where the slightest error could mean the difference between life and death. When there is no explanation of how a model makes a decision, it is hard to place trust in its decisions, let trust that it can make decisions better than human experts can.

Many advanced machine learning models are typically black boxes. In a black box model, we might control the data, the input, and the end goal or the question to tackle. However, we do not have a clear vision of the steps that produce a final output or the prediction in the case of supervised learning. This becomes problematic, especially for deep learning models, as these models' architecture consists of many hidden layers of nodes that adjust weights through pattern recognition. Explainable Artificial Intelligence (XAI) is a research area that proposes a set of tools and frameworks to facilitate an understanding of the logic that goes into machine learning models to arrive at their predictions. XAI allows the interpretation of model decisions and the most critical contributors to particular predictions, ultimately increasing transparency and trust in AI. For instance, in medical imaging with CT scans or MRIs to detect cancer cells, XAI can generate saliency maps highlighting regions or pixels the model uses to identify abnormal cell growth [33, 16]. XAI is also helpful for debugging models, identifying sources of bias, and addressing ethical issues. For example, it allows for troubleshooting a model's performance, explaining consistent misclassifications, such as when a model erroneously associates a pothole in the road with a traffic violation. Additionally, XAI can reveal learned biases, such as gender bias against women in recruitment algorithms, when a model is trained on employment data from a male-dominated workforce [41].

While the potential applications of XAI are broad, our primary focus is on a specific XAI method known as *instance attribution* (IA) and its applicability within the Information Retrieval (IR) domain. Instance attribution, also known as *instance-based explanations*, is a subset of XAI methods that focuses on explaining the models in relation to the training data. Instance attribution, when applied to a prediction for a test example, identifies the most influential training instances contributing to that prediction. For instance, in the case of the model classifying illegal turns from traffic surveillance camera data, a natural way to explain this model misprediction is by providing examples supporting this misprediction. Using instance attribution, the examples most relevant to this prediction may have a common property: for instance, they all contain a pothole in the image data that causes the model to mistakenly associate it with traffic violations.

Query: ....is considered the father of modern medicine.

Rank	Score	Label	Passage
1	0.61	1	TRUE. Hipocrates is considered the father of modern medicine because he did not believe [...]

Most influential training data:

Label	Query	Passage
1	.... is the color of the longest [...]	The color red is the longest wavelength in the visible light spectrum [...]
1	..... is the law that prevents [...]	President Trump reportedly vented to Secretary of State Rex Tillerson [...]
1	....., the leading philosopher of [...]	Sartre's activity as a playwright, novelist and literary critic gave [...]
1	...... are tradition bound, suspicious [...]	The late majority are skeptical—they adopt an innovation only after a [...]

Figure 1.1: An example of most influential training data for a ranking model prediction using TracInCP.The top portion is the test query and passage pair with relevance label 1 and model prediction of 0.61. The bottom portion is the most influential training data for this test instance. The influential training instances share the same query structure as the test instances, beginning with "..." and passages are the completion of missing parts in the queries.

One instance attribution approach for identifying relevant examples for a prediction is using *influence functions* (IF) to quantify the impact of each example. A prominent method for computing a scalar value for influence in this context is TracInCP. [59]. Figure 1.1 showcases an example application of TracInCP. In this example, the ranking model predicts a passage for the test query as relevant where the true ranking of this pair 1. For this scenario,m using influence functions helps us understand the reasons for this model prediction. A closer examination of the most influential data for this test pair shows that the most influential training examples are very similar in query formulation and structure, all beginning with "...".

Instance attribution methods have demonstrated success in various domains, including deep learning, such as image classification, object recognition, and adversarial image generation [42, 89, 59], as well as in recommender systems [17] and natural language processing (NLP) [32, 57]. However, their full exploration in the context of information retrieval (IR) tasks remains limited. An instance attribution task in document retrieval context can provide valuable insights into query and document text representations we little know about in complex and over-parameterized transformer models commonly used for the task [83].

Instance attribution in a document retrieval context can offer valuable insights into the representations of query and document text within complex and overparameterized transformer models commonly employed for this task. These dense neural rankers are often pre-trained on extensive corpora and fine-tuned for specific retrieval objectives. However, these overparameterized models may learn shortcut patterns that do not align well with human understanding [60, 28, 83]. Therefore, it is essential to determine whether the patterns learned by the models are genuinely meaningful and not mere shortcuts. By applying instance attribution to the document retrieval task, we can address questions like, "Which queries or documents in the training set influence the predicted relevance of a document retrieved for a test query?". Furthermore, specialized search systems, such as those used in legal, medicine, journalism, and patent searches, require transparency, control, agency, and traceability of search results [4].

Current approaches in explainable AI frequently focus on generating explanations for individual training examples. The impact analysis performed within these approaches often involves either systematically eliminating problematic instances one by one or generating adversarial examples. There needs to be more exploration of methods to explain the functioning of deep retrieval models or to debug datasets used to train these models using instance attribution. Answering the question, "Which training data led the model to make this decision?" is particularly important. In supervised learning, where the model is trained on labeled data, controlling the training data input to the model is one of the main factors determining the quality of a deep learning model. Given the structural complexity of dense retrieval models, just as any deep learning model in general, trained on massive datasets, manual inspection of the influence of every data point is beyond conceivable. The primary motivation for this thesis is to understand how individual examples and subsets of training data influence the learning of a deep neural ranker and whether we can create smaller and, therefore, more efficient datasets for ranking tasks using instance attribution.

The contributions of this thesis are:

1. We apply the instance attribution method TracInCP, an approximation of Influence Functions (IF), to the passage re-ranking setting to infer the influence of query-passage training data on ranking model predictions.
2. We propose a training data subset selection approach based on influence and conduct experiments

where we train ranker model on different subsets and compare their performance to baseline rankers.

3. We also evaluate the influence-based subset selection approach's generalization capabilities. Our findings indicate that reduced training data does not significantly harm ranker model performance. However, we conclude that training the model with smaller influence-based subsets does not perform significantly better than baseline models.
4. We perform a qualitative human inspection to explain and identify patterns within influential and pruned training examples. We investigate what makes the model mis-classify a test example, looking at training examples supporting and harming that prediction and group queries by context or common characteristics.
5. We investigate the factors that lead to a particular document being ranked higher for a query than another document in the passage ranking setting. This investigation also reveals potential reasons why the influence-based subsets selection approaches did not perform as expected.
6. Finally, we analyze training data self-influence to showcase another use case of instance attribution: diagnosing problematic training examples, such as outliers and mislabeled instances. We manually annotate the top 300 high self-influence training examples and a random 300 training example. We demonstrate that using self-influences for cleaning datasets could be a viable application of instance attribution.

The thesis is structured as follows: Chapter 2 offers essential insights into information retrieval and provides the technical background necessary for comprehending the experiments conducted in the subsequent chapters, found in Section 2.1. Section 2.2 focuses on the general concept of interpretability in AI and its neural network interpretation. Chapter 3 provides an overview of related work on instance attribution, categorizing it by application purpose. It also reviews existing work on instance attribution within the context of information retrieval. Chapter 4 is dedicated to presenting the results of experiments conducted for subset selection with instance attribution. Chapter 5 presents a qualitative analysis of the selected instance attribution method and explores the model explanations derived from this analysis. Chapter 6 serves as the conclusion, providing a collective analysis of the results, offering potential explanations for the outcomes, and suggesting avenues of improvement for further research in this task.

.

2

Context and Background

In this chapter, we first provide background information to familiarize the readers with the concepts that are the foundation of this thesis, namely Information Retrieval, Explainable Artificial Intelligence, and instance attribution.

2.1. Information Retrieval

Information Retrieval(IR) is about finding information in a collection. Despite its seemingly broad scope, this is an activity in which people regularly participate. The IR principles outlined in this paper are relevant to various information items, including books, documents, images, audio clips, video clips, and more, all involving retrieving intellectual content. However, the primary emphasis of this work is on the mainstream practice of information retrieval, which pertains to the description and recovery of written text. As a result, the subsequent concepts and methods of information retrieval discussed in this chapter will fall within the textual context.

In Computer Science, IR focuses on handling document collections with free text, often termed “unstructured” data, to quickly and accurately search for the desired information within text-based queries. “Unstructured data” refers to information lacking clear computer-friendly organization [47]. Relational databases are a prime example of this type of organization. Databases require structured data that adhere to a defined structure to employ query languages with formal semantics such as regular expressions, SQL statements, and relational algebra expressions and yield precise matches, leaving no room for partial or relevant matches. In IR, the primary focus lies in recovering relevant documents, even if they don’t precisely match the query. This is particularly true because free text rarely aligns exactly with the query.

2.1.1. Evaluation Metrics in IR

The standard protocol for assessing the efficacy of an information retrieval system involves utilizing a test collection. The collection involves a corpus of documents, a suite of information needs articulated as queries, and judgments reflecting document relevance —typically binary -categorizing documents as relevant or non-relevant. Various evaluation metrics have been employed to evaluate the performance and effectiveness of information retrieval systems, each providing unique insights into various aspects of retrieval quality. This section begins with a discussion of the concept relevance in IR followed by an examination of commonly employed metrics and their relevance to the present research.

The Concept of Relevance

The standard approach to evaluating the quality of an IR system revolves around the notion of relevance. In essence, determining a result’s relevance depends on whether it can address the underlying need for information rather than merely aligning with the query terms. This distinction between information need and query is noteworthy, as a query might not always explicitly reveal the true information need. For example, consider the query ’stained couch’ used in a search engine. This query arises from the latent information needed to discover methods for effectively eliminating a stubborn stain from furniture.

If the query solely dictated the relevance, a document merely satisfying the literal word composition of the query could erroneously be labeled as relevant.

In fact, this mirrors how humans assess the relevance of information. This insight leads to a second point: the inherent subjectivity of relevance. Even when two users share an identical information need, their assessments of a document's relevance may diverge due to multiple factors. The documents retrieved and presented to a person at a particular time could shape their assessment of subsequent documents showcasing the dynamic nature of relevance over time. Volume, too, can cause a shift. When individuals are presented with numerous strongly relevant examples, this exposure can lead them to categorize specific documents as irrelevant or less relevant compared to other robust results. It's worth noting that such an assessment might not hold if the number of documents was much less. Alternatively, when combined with an individual's present context, considerations like credibility, specificity, recency, and clarity [15] of the result can collectively shape a perception of relevance beyond just the content.

Given these complexities, the gold standard or ground truth for relevance judgment is established in relation to a user's information need by human judges. This usually involves a binary classification of documents in a test collection (e.g., [79]) as relevant or non-relevant. Adequate size is imperative for the test document collection and information suite, as performance is averaged across substantial test sets, and outcomes can exhibit notable variability across diverse documents and information needs. The rule of thumb is a minimum of 50 information needs (queries) [47] to ensure sufficient coverage.

Recall, precision, and accuracy

In an ideal scenario, when a query is submitted to an IR system, the retrieved documents decided by the system would exclusively consist of relevant ones as indicated by the ground truth. However, in reality, IR systems often also retrieve many non-relevant documents. Point-wise metrics precision and recall and several derivative summary metrics can be used as fundamental yet suboptimal metrics for evaluating the effectiveness of a retrieval system given a test collection. Given the following contingency table,

	Relevant	Not Relevant	Total
Retrieved	A	B	A+B
Not Retrieved	C	D	C+D
Total	A+C	B+D	A+B+C+D

Table 2.1: Contingency table demonstrating the partition of the total document set into relevant and non-relevant categories given their retrieval status (retrieved and not retrieved). The number of relevant and retrieved documents is A, relevant and not retrieved is B, non-relevant and retrieved is C, and non-relevant and not retrieved is D. Precision, recall, and accuracy can be calculated using the bivariate counts in rows and columns.

Precision is the proportion of relevant items amongst retrieved items.

$$Precision = P(Relevant \mid Retrieved) = \frac{A}{A+B} \tag{2.1}$$

Recall represents the ratio of retrieved items among all relevant items.

$$Recall = P(Retrieved \mid Relevant) = \frac{A}{A+C} \tag{2.2}$$

Accuracy is the fraction of the correctly classified items by the system, either relevant or non-relevant.

$$\begin{aligned} Accuracy = P(Relevant \cap Retrieved) + P(Non - Relevant \cap NotRetrieved) \\ = \frac{A+D}{A+B+C+D} \end{aligned} \tag{2.3}$$

Although accuracy is commonly favored as the evaluation metric for numerous machine learning classification tasks, it is not as appropriate for evaluating IR performance [73]. Most tasks in IR, such as document retrieval, can be interpreted as a classification task, categorizing documents as relevant

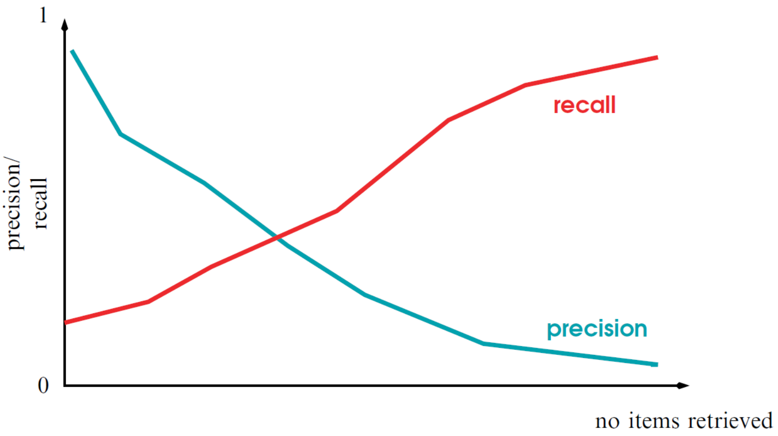


Figure 2.1: The inverse relationship between precision and recall, figure from [73]. Increased recall means most documents will be retrieved, potentially also irrelevant documents alongside the relevant ones. This dilutes the overall precision, thus the precision curve in blue has an inverse relationship with the recall curve in red.

or non-relevant. However, accuracy proves inadequate for assessment due to the highly imbalanced data distribution, predominantly skewed toward the non-relevant category [47]. As a result, a system that seemingly performs high in accuracy could be misleadingly attributed to its handling of numerous non-relevant items, overshadowing its performance with relevant items that are truly significant but fewer in number. Precision is an indication of the quality of the answer set. But this does not consider the total number of relevant documents. A potential issue with recall is that recall will increase if we retrieve most documents, but this does not necessarily mean that the retrieved set is highly relevant or accurate. As the retrieval system aims to capture as many relevant documents as possible, it might also include a significant number of irrelevant documents. This can lead to a dilution of precision, where the proportion of truly relevant documents in the retrieved set diminishes. Figure 2.1 demonstrates this inverse relationship between precision and recall.

(Mean) Reciprocal Rank

Reciprocal Rank (RR) places more emphasis on the initial encounter of a relevant document, particularly suited for scenarios with sparse judgments. This approach assumes a single relevant document or the user's contentment with the highest-ranked item. For a given query set Q and $FirstRank$, the rank of the first relevant document for query $q \in Q$, RR is determined by computing the weighted average of reciprocal ranks.

$$MRR(Q) = \frac{1}{|Q|} * \sum_{q \in Q, i=1} \frac{1}{FirstRank(q)} \tag{2.4}$$

NDCG

Normalized Discounted Cumulative Gain (NDCG), a popular metric for ranking quality of search engines, recommendation systems and other information retrieval systems, addresses scenarios involving non-binary interpretations of relevance. This metric grades the relevance of results taking the positional information [36] of the retrieved items into account. For a query document pair, the discounted cumulative gain (DCG) of a single document D is calculated as:

$$DCG(D) = \sum_{d \in D, i=1} \frac{rel(d)}{\log_2(i+1)} \tag{2.5}$$

where $rel(d)$ is the gain, in other words, the relevance value for a single query-document pair, and the denominator is the position discounting. Then, nDCG for the set of all queries Q is the normalized ratio of the actual results to the ideal sorting, in other words, the ground truth.

$$nDCG(Q) = \frac{1}{|Q|} * \sum_{q \in Q} \frac{DCG(q)}{DCG(sorted(rel(q))} \tag{2.6}$$

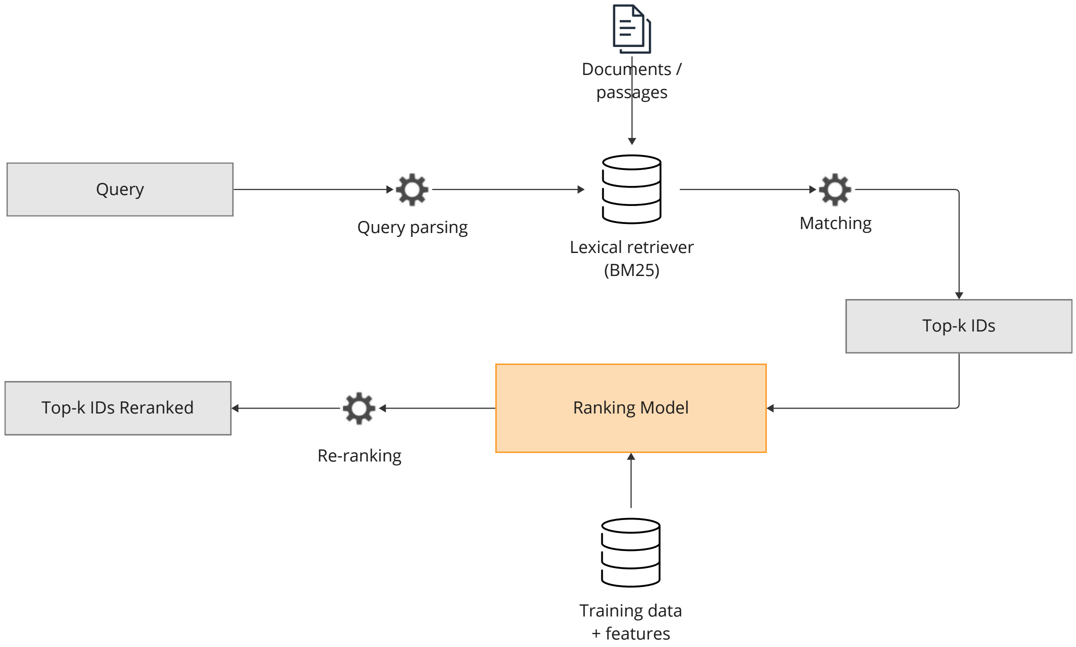


Figure 2.2: An Overview of the Document/Passage Reranking Process. Queries are parsed and an initial topk-k list ranking possibly relevant documents for each query is constructed with BM25. At the second stage a more complex ranking model is employed to re-rank the retrieved top-k candidate documents.

2.1.2. IR Tasks

The IR domain spans a diverse array of tasks, each potentially involving many methodologies, metrics, and datasets specific to the task. Despite this diversity, the fundamental objective remains consistent: efficiently retrieving relevant information from extensive repositories. One pivotal IR task is ad-hoc retrieval, which aims to rank relevant documents from a web corpus to place the most relevant documents to user queries at the top [22, 30]. In this context, document retrieval is a well-established subtype wherein systems aim to return entire textual documents as output. Passage retrieval is another specialized task within ad-hoc retrieval, gaining popularity in response to the growing prevalence of question-answering systems where the trend is extracting relevant information from shorter textual segments [23]. This sub-task concentrates on retrieving concise passages that contain answers to a given query. Entity retrieval involves finding documents discussing specific entities, commonly used in tasks like expert finding [71, 7]. Temporal retrieval [14] is concerned with retrieving temporally relevant information, whereas geospatial retrieval [39] focuses on retrieving location-based information. Finally, multimedia retrieval expands to images, videos, and audio, facilitating content-based search across different media types.

Passage Re-Ranking

This study focuses on text-based document retrieval, particularly reranking text documents presented as short passages in the MSMARCO passage dataset. The ranking model generates a prioritized list of documents, with the highest item presumed to be most relevant to the given query. The general workflow of this two-fold text retrieval process is depicted in Figure 2.2.

In the first stage, a large set of possibly relevant documents to a given query are obtained from a corpus through a standard mechanism, often BM25 (Section 2.1.3).In the second stage, the re-ranking phase, retrieved documents are scored and re-ranked. During this phase, it is common to employ computationally intensive methods, such as Learn-to-Rank models (LTR) or neural rankers such as BERT-Reranking [53].

Given a query, the ranking model outputs a ranked list of documents so that the top-ranked items should be more relevant to the user's query. A general flowchart of the two-fold text retrieval process is illustrated in Figure 2.2. A large collection of documents is indexed for fast retrieval. Given a text-based query from the user, candidate documents are obtained from an unsupervised ranking stage, such as BM25 which uses the initial set of indexed documents and the query as inputs. During this first ranking

Query id: 1136987
Query: Why did dalton think it was important to use his system of symbols for the chemical elements?
Passage id: 2829865
Passage: It was essential to remind people that all the matter around us has one base unit: the atom. Dalton thought it was important to use his system of symbols for the representation of chemical elements because it was helpful to remember exactly which atoms lay at the basic structure of the matter. It was easy for people to use symbols for elements in chemical formulas. He pictured the atoms as small balls and this is the same way that he depicted models of elements and compounds.
Relevance: 1

Query id: 1136987
Query: Why did dalton think it was important to use his system of symbols for the chemical elements?
Passage id: 8017828
Passage: Two weeks before the season opener against Baylor, TCU head coach Gary Patterson named Dalton the starter. Dalton was named the 2007 Texas Bowl MVP in TCU's 20–13 victory over Houston. After going 8–5 as a freshman, he accumulated a record of 34–3 as a starter for the rest of his career at TCU.
Relevance: 0

Figure 2.3: Example query passage pairs from MSMARCO passage train set. Both examples contain the same query and different passages as pairs. The first example (left) demonstrates a relevant query passage pair and the second (right) demonstrates a non-relevant passage paired with the same query.

phase, recall is more important than precision to cover all possible relevant documents and forward a set of candidate documents that has both relevant and irrelevant documents to the commonly neural based re-ranking stage. The output of the ranking model is a set of relevant documents to the user's query which are returned in a particular order.

MSMARCO Passage Dataset

MS MARCO, standing for Machine Reading Comprehension[52] is a collection of large-scale IR datasets curated for machine reading comprehension (MRC), question answering (QA), passage ranking, keyphrase extraction, and conversational search studies tasks. The data collection was created by sampling and anonymizing Bing and Cortana's click logs to reproduce real-world scenarios of Web search, resulting in noisy annotations. The authors' motivation to create this dataset was to provide a large enough dataset that could be used for training deep neural models, a property existing datasets for MRC and QA datasets were lacking at the time. The passage ranking task is formulated based on the questions in the Question Answering Dataset in the collection and the passages extracted from documents retrieved by Bing in response to the questions. MSMARCO Passage Dataset, one of the datasets in the MSMARCO datasets collection, was created by asking human annotators to mark passages in documents they have used to construct answers to the questions. According to the authors, the collection of documents is very large and noisy, and not all relevant passages are necessarily annotated.

The authors argue that, in contrast to the existing datasets for Machine Reading Comprehension (MRC) task, the questions in MSMARCO collection offer a more accurate representation of a natural distribution of users' information needs. As the questions in MSMARCO Passage dataset are actual search queries of Bing users, the dataset captures the messy structure of human input, such as typos, abbreviations, grammar mistakes, or very brief queries. In contrast, datasets preceding MSMARCO are often synthetic, where the questions are constructed by crowd workers based on passages provided to them. Such datasets often contain high-quality text which may not fully capture the messy nature of real-world user queries submitted to search engines like Bing. One of the proposed tasks for the MSMARCO Passage dataset by the authors, which is also the task focused on this thesis, is the passage re-ranking task for which a system is provided with a question and a set of 1000 retrieved passages using the BM25 model [66]. The goal is to re-rank passages in descending order based on their relevancy to the question content.

	Dataset split				
	train	train triples v2	dev	trec-2019 judged	trec-2020 judged
# queries	808,731	808,731	101,093	43	54
# qrels	532,761	532,761	59,273	9,260	11,386
# docpairs		397 M			

Table 2.2: The number of queries and relevance labels for query document pairs for the dataset splits and total document pairs in train triples set.

MSMARCO passage dataset contains 8,8 M passages and 1M queries extracted from a corpus of 3M documents, roughly 400M triplets, and a set of relevance labels per query and passage id pair. The set of relevant passages are constructed by human editors, who annotated the passages they used to compose an answer to the given query. The dataset is divided into training, dev, triples and various evaluation set splits. In this thesis we use two of these evaluation set splits: trec-2019 judged and trec-2020 judged. These test splits are a subset of queries filtered by NIST assessors [1] from the evaluation set of TREC Deep Learning (DL) 2019 and 2020 shared tasks [20, 21]. The triples split contains additional document pairs data, which provides triplets instead of pairs: per query id, one relevant and one non-relevant document based on BM25 scores of the documents. Table 2.2 summarizes the number of queries, qrels, and document pairs of these splits.

2.1.3. Traditional IR Models

Traditional Information Retrieval (IR) models encompass a range of approaches, including Boolean retrieval models, vector space models [69], probabilistic retrieval models [65], language modeling and Learning-to-Rank (LTR) [46, 44]. Each of these models possesses distinct approaches that could be explored in detail. However, it's important to clarify that the primary emphasis of this thesis does not revolve around traditional IR models. Instead, the intention is to provide concise overviews of a select few, namely boolean models, TF-IDF, a vector space model, and BM25, a probabilistic model, a relevant concept in sections describing the experiments.

In most retrieval scenarios, linearly scanning many documents is impractical. A binary term-document incidence matrix is a data structure that represents the presence or absence of terms in documents using a matrix and enables quicker retrieval. In this context, a "term" signifies an indexed unit, often corresponding to a word. A "document" represents the indexed retrieval unit, often paragraphs or documents. A "collection" refers to a group of documents over which the retrieval system operates, commonly called a corpus.

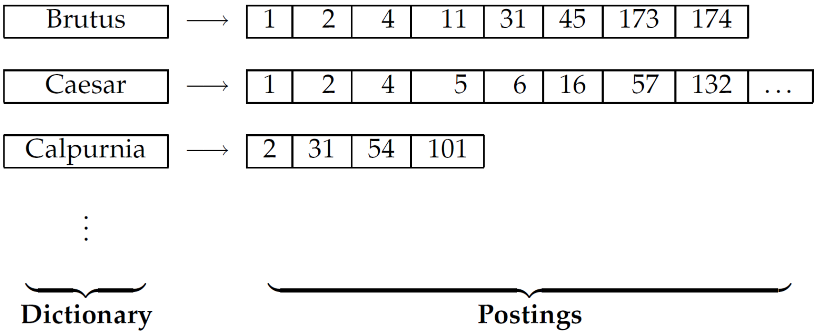
Considering the dimensions of a term-document incidence matrix, which corresponds to the number of unique words across documents, it becomes evident that such a matrix is both sparse and memory-intensive for computer storage. The **inverted index** overcomes this limitation. An inverted index maps a term to the documents containing it. Terms are stored in a dictionary, while postings comprise a list of document IDs containing the term. The inverted index is also binary, indicating whether a document contains a word. **Boolean models** operate similarly to a term-document matrix. However, it's important to note that these models primarily indicate whether a document contains specific query terms. If a Boolean model were applied to retrieval, it would yield a set of documents that fulfill the specified Boolean condition. A Boolean query lacks the sophistication of natural language, and consequently, there would be no means to establish a hierarchy or ranking among the retrieved documents based on their relevance.

Vector space models introduce a ranking approach predicated on relevance scores computed for each document. This method treats every document as a vector, wherein each component represents a term's weight. The TF-IDF weight calculation forms the basis for assessing how relevant a given query is to each document in the collection. TF-IDF (eq. 2.9) is computed for a document by multiplying two values: the number of times a word appears in the document (term frequency, eq. 2.7) and the rarity of the word across all documents (inverse document frequency, eq. 2.8). Term frequency can be as simple as counting how often a word appears in the document. Inverse document frequency shows how common or uncommon a word is in all documents. A value closer to 0 indicates a more common

[1] https://trec.nist.gov/data/deep2019.html and https://trec.nist.gov/data/deep2020.html

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Antony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
...							

(a) An example of term-document incidence matrix



(b) An example of inverted index

Figure 2.4: Comparison of term-document incidence matrix (a) and inverted index (b)[47]. The term document incidence matrix represents the presence of terms in documents with Boolean indicators. Inverted index stores terms as keys of a dictionary structure and maps terms to the documents in a list containing IDs.

word. It's calculated by dividing the total document count by the number of documents containing the word, with a logarithm. Multiplying these produces TF-IDF weight for a word t in a document $d \in D$, reflecting its relevance.

$$tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \tag{2.7}$$

$$idf(t,D) = \log(\frac{N}{count(d \in D : t \in d})) \tag{2.8}$$

$$tfidf(t,d,D) = tf(t,d) * idf(t,d) \tag{2.9}$$

With TF-IDF, each document and query can be viewed as separate vectors. A document vector's elements correspond to TF-IDF weights of terms. Subsequently, determining relevance scores entails calculating the similarity between the two vectors representing a query and a document. A widely-used metric for this purpose is cosine similarity, quantifying the extent of similarity between the query and document vectors within the vector space. This vector space model provides a method for ranking documents based on their relevance to user queries.

Probabilistic models leverage probability theory to represent and evaluate the relevance of documents in response to user queries. These models operate on the premise that a document's relevance to a query can be quantified through a probability distribution, reflecting the likelihood that users perceive the document as relevant given the query. Noteworthy examples of such models include the **Binary Independence Model (BIM)**, which assumes term independence, the **Language Model for Information Retrieval (LMIR)**, treating queries and documents as probabilistic language models, and the **Probabilistic Information Retrieval Model (PIRM)**, emphasizing relevance estimation using probabilistic reasoning.

A prominent model within this context is the **Best Matching 25 (BM25)** [66], considered a heuristic probabilistic model. As a ranking mechanism, BM25 assesses document relevance to a query by considering the frequency of query terms within the document and the document's length. BM25 generates sparse vectors based on term frequency and inverse document frequency. Its advantages

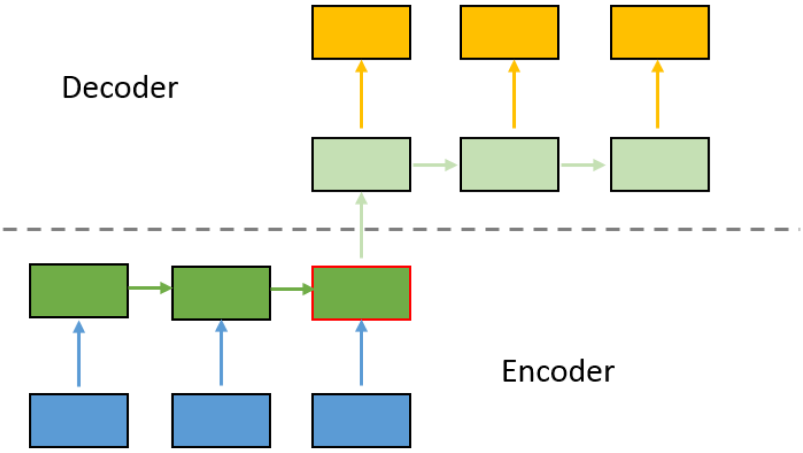


Figure 2.5: The encoder and the decoder[80]. The rectangle surrounded by the red box is the input of the decoder. The first row of rectangles represents the input items, and the second row represents the hidden states of the encoder. The first row of rectangles represents the hidden states of the decoder, rectangles in the second row are the output items.

include speed, efficient indexing, and requiring no training. However, it has limitations, such as potential lexical gaps and the inability to maintain word order. Consequently, BM25 is often employed as a primary-stage retrieval model to leverage its strengths within information retrieval tasks.

2.1.4. Neural IR Models

Information retrieval has experienced a transformative leap with neural ranking models following the rise of deep learning techniques. This evolution is driven by the remarkable capability of neural networks to learn representations of text directly from raw text inputs [49]. In contrast to the constraints of traditional methods, which relied on hand-crafted features tailored to specific tasks, neural models offer a more flexible and data-driven approach to IR. This makes them well-suited to handle complex tasks involved in relevance estimation during ranking. Additionally, these models are better equipped to handle the vagueness and complexity of relevance judgments. Although traditional LTR models achieve good performance in various IR applications, **dense retrieval models** have become state-of-the-art today, especially in more challenging tasks like cross-lingual retrieval [76, 37], open domain QA [40, 85], conversational search [45, 90]. There are different neural approaches in IR. A neural network can be used only at the point of matching, or it can be the focus on learning effective representations of text.

While neural models can be applied to a broad spectrum of tasks, the focus of this section will be confined to neural ranking models in the context of textual retrieval.

In contemporary IR research, employing large language models (LLMs) as the foundation for dense retrieval is common practice. LLM variants like BERT (Bidirectional Encoder Representations from Transformers), GPT(Generative Pretrained Transformer) transformed the comprehension, processing, and retrieval of textual data. The remarkable ability of these models to capture semantic nuances and intricate contextual relationships aligns well with text retrieval tasks in IR. Dense retrieval operates on the principle of representing queries and documents in high-dimensional vector space - or *embeddings* - allowing assessment of their semantic relevance using similarity metrics, for instance, cosine similarity. LLMs like BERT construct dense embeddings that contain contextual information for individual words and phrases, allowing a more nuanced semantic representation of text rather than mere keyword overlap seen in traditional retrieval methods or hand-crafted features in LTR [88]. In the following section, we will provide a background on transformers, the predecessors to models like BERT and other LLMs, and the role of transformer-based models in classifying text sequences, an integral part of this study's approach.

2.1.5. Sequence Classification with Transformers

Transformers mark a significant leap toward the advanced pre-trained embedding models that are prevalent today. While preceding models (e.g. Word2Vec [48], Glove [56]) were capable of representing words as vectors, these vectors lacked contextual information about the words in a sequence. In natural

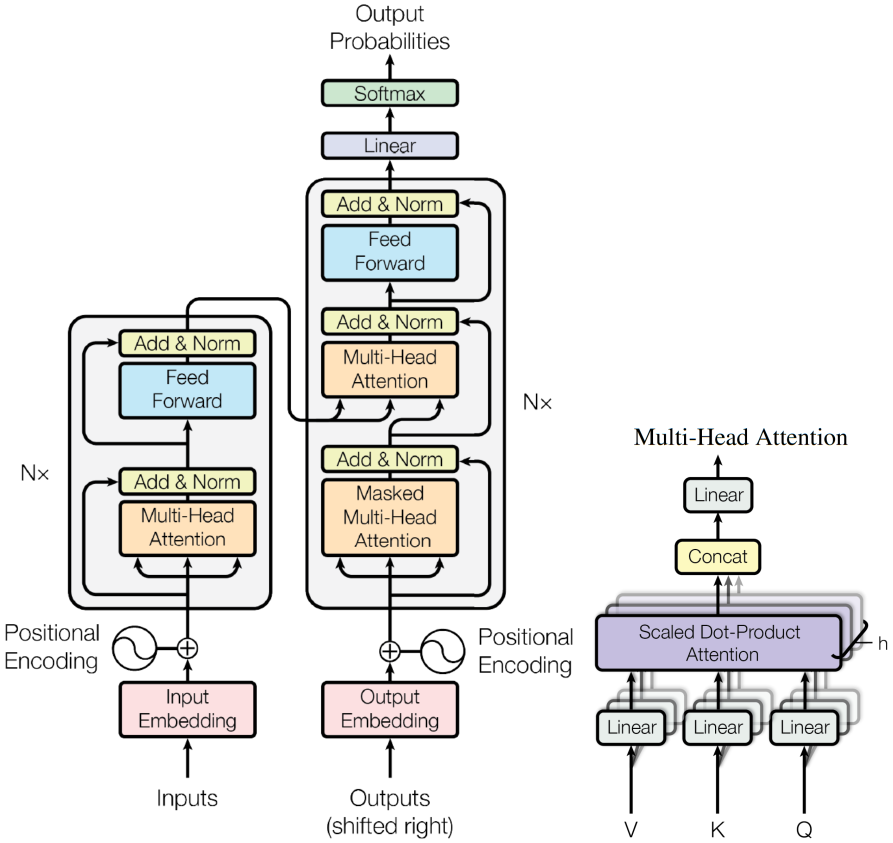


Figure 2.6: The transformer model architecture (left) and multi-head attention (right), figures from [78]. The transformer model uses positional encodings, self-attention, and multi-head attention. Positional encoding allows incorporating the sequence order by introducing sinusoidal activation to each input embedding based on its position. The self-attention mechanism applies the attention mechanism to a sequence, which learns the inter-dependency between a given word and the previous part of the sequence. The multi-head attention block computes multiple attention-weighted sums instead of single attention passes over the values, which allows the representation of several sets of relationships within a sequence.

language, it is a prevalent phenomenon that the meaning of a word can diverge based on its surrounding context. Before transformers, the word 'bank' would receive identical encoding in sentences 'People were having a picnic at the river bank' and 'He had to go to the bank'.

Prior to transformer-based encoder-decoder [78], Recurrent Neural Networks(vanilla RNNs)[68] and later its variants long short-term memory network (LSTM) [34] and gated recurrent unit (GRU) [18] were the main choices for processing sequences of text. An encoder-decoder architecture is commonly employed in scenarios where the input consists of a data sequence, and the desired output is another sequence. Known as sequence-to-sequence [72] modeling, this approach finds its applications in tasks like machine translation. Figure 2.5 illustrates an RNN model for the sequence-to-sequence problem.

The encoder-decoder architecture comprises three key components. The encoder encompasses a stack of recurrent units, such as RNNs, LSTMs, or GRU cells. Each unit processes an individual element from the input sequence, accumulates information associated with each element in the sequence, and propagates the information forward. The output of this encoding process is the final hidden state, the rectangle surrounded by the red box in Figure 2.5, referred to as the encoder vector. This vector encapsulates the information from all input elements of the sequence and serves as the initial hidden state for the decoder module. The decoder decodes the hidden representation for the relevant task. In sequence-to-sequence problems, there is typically a relationship between the elements in the output and input sequences. However, in a conventional RNN model, generating a fixed-length hidden state prevents the model from assigning different weights to individual items within an input sequence in a discernible manner [80]. As a result, regardless of the specific output item under consideration for prediction, all items in the input sequence are equally important. This creates an information bottleneck, as the information generated across various words in a sequence is passed through a single connection point to the decoder without accounting for importance weights. This limitation led researchers to integrate the attention mechanism.

As showcased in transformer models and BERT, the attention mechanism enables the selective focus on relevant details, thus reducing information overload. The transformer model (Figure 2.6) bypasses the usage of RNN units by leveraging positional encodings, self-attention, and multi-head attention. Positional encoding allows incorporating the sequence order by introducing sinusoidal activation to each input embedding based on its position. Self-attention mechanism applies the attention mechanism to a sequence, which learns the inter-dependency between a given word and the previous part of the sequence. The multi-head attention block computes multiple attention-weighted sums instead of single attention passes over the values, which allows the representation of several sets of relationships within a sequence.

Pretrained models

The revolutionary advantage of transformer architecture is its capacity to leverage the core of a model while replacing the last few layers to suit different tasks. Referred to as pre-trained models, these models showcase remarkable efficacy in learning feature representations across a spectrum of tasks, including text classification and generation. Pre-training entails training a general model with a large corpus on multiple tasks that can be fine-tuned easily in different downstream applications [91]. In IR and NLP, most models established as performance and processing speed benchmarks across diverse tasks rely on pre-trained models that have been fine-tuned for smaller-scale tasks. While the methodologies for establishing the relevance between queries and passages vary, contemporary works in passage retrieval frequently utilize a pre-trained transformer model to generate sentence embeddings as the primary step.

BERT There are several variations of transformer models, each designed with distinct components. For instance, some models integrate the encoder and decoder components from the original architecture, as seen in BART [43]. Others exclusively employ a decoder, like GPT-2 [62] and GPT-3 [12], while some solely utilize an encoder, such as BERT.

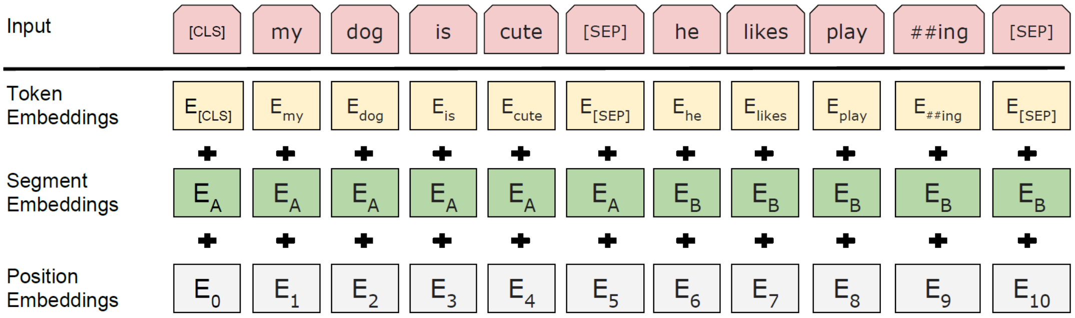

Input
[CLS]
my
dog
is
cute
[SEP]
he
likes
play
##ing
[SEP]
Token Embeddings
$E_{[CLS]}$ E_{my} E_{dog} E_{is} E_{cute} $E_{[SEP]}$ E_{he} E_{likes} E_{play} $E_{\#\#ing}$ $E_{[SEP]}$
Segment Embeddings
E_A E_A E_A E_A E_A E_A E_B E_B E_B E_B E_B
Position Embeddings
E_0 E_1 E_2 E_3 E_4 E_5 E_6 E_7 E_8 E_9 E_{10}

Figure 2.7: BERT input embeddings, figure from [25]. BERT represents an input sequence with three distinct types of embeddings: token, segment, and position embeddings. Token embeddings are pre-trained and generated by indexing a matrix with sizes in line with vocabulary size and the number of hidden layers in the model. The positional embedding vector contains word position within sentences. Segment embeddings are used for marking which sentence an embedding belongs to.

BERT stands as one of the prominent transformer models that have been trained extensively. Originally released, BERT was developed in two distinct variations. The first model, $\text{BERT}_{\text{BASE}}$, comprises 12 stacks of transformers with 110 million parameters. The second, $\text{BERT}_{\text{LARGE}}$, comprises 24 stacks of transformers with a parameter count of 340 million. BERT can accommodate lengthy input contexts as it has been trained on massive data, including the entire Wikipedia and books corpora in a multi-task-objective manner. The first pre-training objective, known as masked language modeling (MLM), involves masking out a fraction (often recommended as 15 %) of input words. The model is then trained to predict these masked words, which aids it in learning contextual cues. The second task, ’next sentence prediction’, involves predicting the subsequent sentence, wherein the model is presented with pairs of sentences. The model can identify relationships between sentences and predict the subsequent sentence through this objective.

BERT's efficiency lies in its capacity to handle extended input contexts. BERT represents an input sequence with three distinct types of embeddings: token, segment, and position embeddings, as shown in Figure 2.7). Token embeddings serve as vectors that encapsulate the essence of individual word tokens within input sentences. By transforming words into vector representations of specific dimensions, token embeddings capture the contextual significance of each word. It's important to note that token embeddings are pre-trained, crafted through the indexing of a matrix sized according to the vocabulary and the number of hidden layers in the model. The vocabulary utilized in this process is derived from a subword segmentation technique known as WordPiece tokenization [84]. This approach initializes new tokens with individual characters from the input and then progressively extends the vocabulary by combining these characters to generate new word units. While token embeddings capture token-specific information, the positional details of tokens are captured with positional embeddings. Position embeddings encode a word's position within a sentence in the form of another vector. BERT's architecture accommodates input sequences of up to 512 characters. The position embeddings layer functions as a lookup table with dimensions (512,768) where each row corresponds to the vector representation of a word at a specific position within the sequence. Segment embeddings play a role in discerning sentence numbers within BERT. These embeddings indicate whether a given token belongs to sentence A or sentence B. A special token [SEP] in the input embedding guides the separation of sentences in the segment embeddings. A dedicated fixed token is assigned to words within each sentence, allowing BERT to interpret sentence relationships accurately.

MiniLM MiniLM[81] has laid a foundational framework for the efficient distillation of LLMs, enabling them to maintain notable accuracy on specific tasks while significantly enhancing inference speed. This process, known as knowledge distillation, involves compressing the extensive knowledge of large 'teacher' model into a more streamlined 'student' counterpart. The initial MiniLM model, MiniLMv1, is a compact variant achieved by distilling prominent teacher models such as BERT and RoBERTa. In this context, the designated teacher models serve as encoder models.

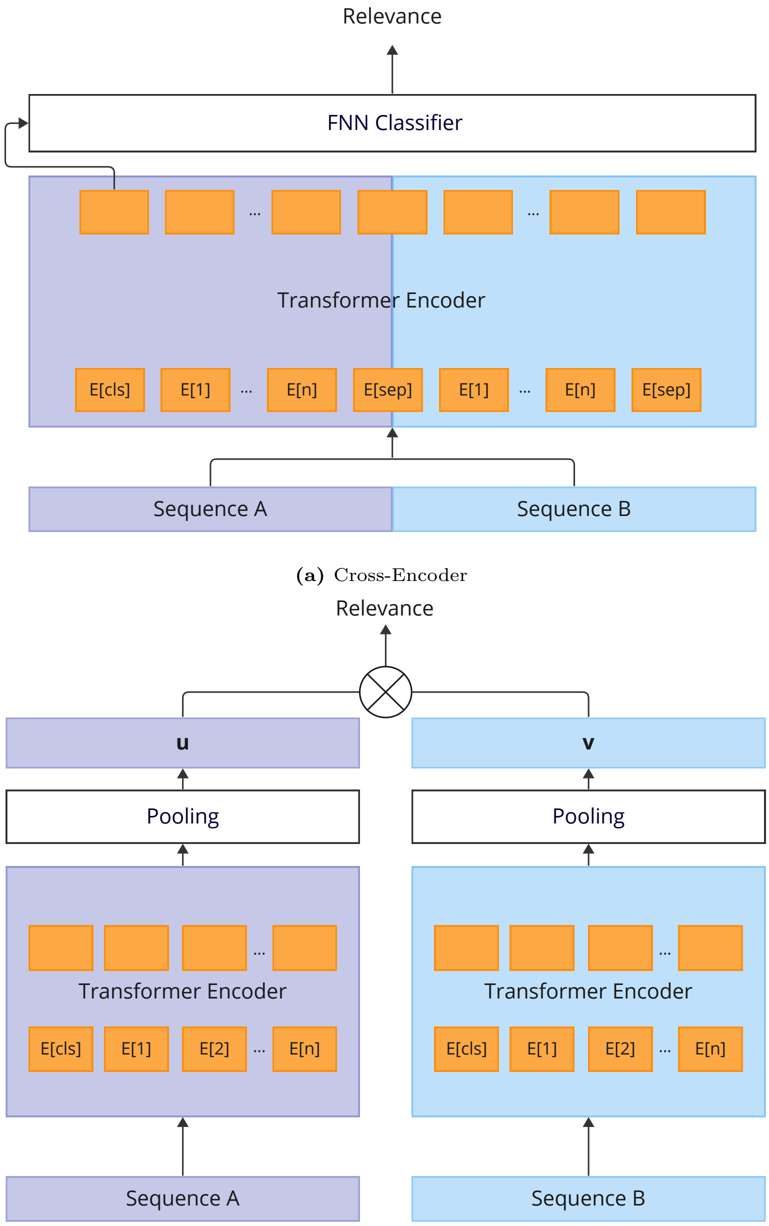
Unlike other distillations of BERT, such as DistilBERT and TinyBERT, MiniLM employs a dual approach. A student model is trained through deep mimicry of the self-attention module within the teacher transformer model's final layer. Additionally, the interrelationships among values in the self-attention vectors serve as a guide for student training. The value relation is determined through a multi-head scaled dot-product calculation, which is then used for computing the KL divergence between the teacher and the student value relations. Using this value as the training objective, the student effectively emulates the teacher's self-attention behavior. MiniLMv2 [82] is an update with greater flexibility over MiniLMv1 by removing the constraint that required student models to possess the same number of attention heads as their respective teacher models.

Bi-Encoder

The bi-encoder model family constitutes a broad class of models that independently map input queries and candidate passages into a shared feature space and often deploy dot product or cosine similarity to quantify their similarity. This category of models includes methodologies like supervised embeddings [6], classical Siamese networks [10], and vector space models [69]. Within the specific context of passage retrieval, which forms the central focus of this study, it's imperative to note that both the query and document are independently generated. This process is illustrated in Figure 2.8b, where a transformer initiates with identical weights for both the query and the document. Subsequently, during the fine-tuning phase, the transformer's weights are permitted to update autonomously for each component. Typically, a pooling operation is commonly employed on both embeddings, resulting in dimension reduction and the creation of distinct vector representations denoted as "u" and "v." In line with this process, a similarity metric is introduced to determine the relevance between a query and a passage vector representation. An example of such metric is cosine similarity, as depicted in Figure 2.8b. Since the generation of passage encodings is decoupled from query inputs, the vector representations of a large fixed passage set can be cached, allowing a faster relevance evaluation phase and larger batch sizes during training to increase performance.

Cross-Encoder

The cross-encoder architecture [63] generates a unified output, typically in the form of a score or label, for pairs of input sequences, commonly sentences, paragraphs, or entire documents. In the context of



(a) Cross-Encoder

(b) Bi-Encoder

Figure 2.8: The structure of conventional transformer encoders used in re-ranking tasks, cross-encoder (a) and bi-encoder (b). The cross-encoder takes the concatenation of sequences (query and passage in passage ranking case) as input to the transformer encoder. The bi-encoder takes sequences (query and passage) as separate inputs to the transformer encoder.

passage retrieval, the generation of embeddings involves concatenating the query and passage texts into an extended input sequence as in Figure 2.8a instead of their separate treatment within a transformer encoder. In each sequence, the initial token is a fixed classification token [CLS], and the ultimate hidden state corresponding to this token serves as the sequence representation, fed into a feed-forward neural network classifier. This process yields a score ranging from 0 to 1, denoting the passage's relevance to the query.

Utilizing a single transformer to produce a jointly encoded representation facilitates a cross-attention mechanism such that each word in the query interacts with every word in the passage within the unified sequence. A key advantage of this is capturing complex interactions among input sequence elements irrespective of length or position. The cross-encoder is robust to domain shift and consistently outperforms bi-encoders, particularly under reduced training data [74]. However, cross encoders are prohibitively slow, as instead of pre-computation, every query passage pair concatenation is done in inference time, followed by a forward pass of the entire model. Therefore, cross-encoders are typically used for re-ranking and not full retrieval. It is also worth noting that cross encoder requires significantly more memory than bi-encoders [77], resulting in a substantially smaller batch size during training [35].

2.2. Interpretability of AI Models

AI models display a remarkable capacity to learn for any task but are difficult for humans to interpret. A prevalent architecture for deep learning, neural networks, can capture complex decision boundaries through iterative refinement of interconnected layer weights and biases. *Backpropagation*, although highly effective, presents difficulties for humans to trace directly. The high number of parameters and depth of models give them state-of-the-art prediction performance but also complicate interpretability. *black-box model* refers to the family of models, including neural networks, whose outputs are not explainable by design. Historically, understanding why these models arrive at certain predictions has not been a priority as long as they yield accurate outcomes. Nevertheless, this lack of transparency becomes a concern when deploying AI in high-stakes decisions such as healthcare diagnoses, credit-risk assessments, autonomous vehicles, and security systems where upholding accountability, fairness, and ethical standards [24] is very crucial.

2.2.1. Explainable Artificial Intelligence

Explainable AI (XAI) is a research branch dedicated to making the decision-making processes of AI systems more understandable and transparent to human users. The central objective of XAI research is to introduce techniques that enhance the interpretability and accountability of AI systems. Trust is an important aspect of XAI. While machine learning is at the forefront of many recent advancements, it's ultimately the trust of users that determines whether a model becomes a useful tool or a part of a product. If users do not trust a model or its predictions, they are unlikely to use it.

An important focus of XAI is the trust of human users. Even though AI is today at the core of many recent advances in science and technology, the users' trust eventually determines whether a model will be used as a tool or within a product. Ribeiro, Singh, and Guestrin describe this kind of trust in two ways: users trusting model predictions enough to act upon them and trusting a model to behave as expected when deployed. An improved understanding of a model reduces the perception of it being a black box, increasing the social acceptance of AI methods.

2.2.2. Taxonomy of XAI methods

There are diverse strategies within the realm of XAI. These strategies exhibit differences in factors such as but not limited to the scope of application, the nature of the AI model, and the intended form of explanation. A common categorization framework highlights the difference between global and local approaches, intrinsic and post-hoc methodologies, and model-specific and model-agnostic strategies [1]. Consequently, it is plausible for an XAI technique to align with multiple classification categories; for instance, an XAI approach might simultaneously encompass global, post-hoc, and model-agnostic attributes.

Interpretation methods are classified as either global or local based on their scope focus. Global methods seek to explain the overall relationship among model outputs, data, and the trained model at a broader level. Global methods aim to determine the average behavior in a deep learning model. A global explanation technique yields an overarching explanation of how a model makes decisions based on its features and learned components, such as weights and parameters. However, this approach demonstrates practical limitations, particularly as the parameter space expands and the model's architecture becomes increasingly complex. The sheer complexity inherent in deep learning models, frequently characterized by millions of weights and parameters, makes the concept of a globally comprehensible explanation unfeasible for human understanding. Consequently, it is a prevalent strategy to decompose it into more manageable sub-modules [51, 24]. Local interpretation methods explain how a single input instance z influences a model's prediction $\hat{y}_z$. Local explanations typically provide insights into predictions for individual instances. However, scaling this approach to provide generalized explanations at the model level necessitates the computation of individual explanations for every prediction, which is both resource-intensive and expensive. The criteria intrinsic or post hoc distinguishes whether the interpretation is achieved by examining the model itself (intrinsic) or by applying methods post-training (post hoc). Intrinsic interpretation is limited to machine learning models that are considered interpretable by design, such as decision trees, rules or linear models [50]. Model families, such as artificial neural networks (ANN), support vector machines (SVM), boosted trees, and random forests, are considered opaque, and their structural complexity prevents users from tracing the logic behind predictions. Post hoc interpretation involves extracting information after an opaque model has been trained. The advantage of this approach is that it does not impact the model's performance as it is treated as a black box [27]. Both Feature Attribution and Instance Attribution are examples of local post-hoc methods, and the latter is the particular focus of this study. In 2.3, we will provide a more detailed discussion of advantages and particular limitations pointed out so far.

The categorization into model-specific and model-agnostic methods differentiates whether the interpretation method depends on the specific model type or operates independently. Although more efficient due to using specific model properties for explanations, model-specific methods lose their applicability when the underlying model is substituted with a different model class. For instance, feature attribution methods based on gradients are model-specific since those methods can only be used with model families trained with gradient descent. Model agnostic methods have the advantage of being applicable to any machine learning model, and they are post-hoc, so they are applied after the model has been trained. These methods operate by analyzing feature input and output pairs. Intrinsically, these methods cannot access internal model components such as weights or structural details. A prominent example of such a model-agnostic method is LIME.

2.3. Instance Attribution

Two well-known attribution methods are *instance attribution* and *feature attribution*. An extensive body of work within the domain of XAI, also in the context of NLP primarily focuses on feature-based explanations [57]. This method focuses on attributing important input features to a particular prediction. The goal of feature attribution is to assign an attribution value per feature. For instance, the set of input features for a text classification setting, where the goal is analyzing sentiment, could be word tokens. Using a feature attribution method, the top tokens, *or features*, with the highest numerical attribution scores for a given sentiment label would correspond to important features for classifying different sentiments [57]. Feature attribution methods provide insights into deep learning models from the perspective of input feature and model prediction relationships. These methods are useful for understanding the effect of features on a model or identifying key input features as a subset for a given task. However, feature attribution methods do not completely explain the relationship between the model and input. The insights gained from these methods are from the perspective of the features, which is only one of many aspects of input data.

Instance attribution methods, in contrast, aim to explain predictions through particular training instances. Unlike feature attribution methods, instance attribution is used for pinpointing examples that either support or oppose a prediction. This relationship between a training example and a prediction is often referred to as *influence*. Instance attribution methods retrieve training samples 'influential' to a given prediction [57]. If the deletion of a data point changes the model parameters or a prediction of a model significantly, that data point is considered to be influential. This change between the prediction of instance j prior to and after the removal of instance i , or influence, can be expressed as in Equation

$$\mathit{Influence}_j^{(-i)} = |\hat{y_i} - \hat{y_j}^{-i}| \tag{2.10}$$

A data point is considered influential We consider a model, ϕ that maps inputs $x_i \in \mathcal{X}$ to targets or labels $y_i \in \mathcal{Y}$. The training set $\mathcal{D} = z_i$ is the combination of inputs and targets such that $z_i = (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$. A prediction made for a test sample is defined as $\hat{y_t} = \phi(x_t)$. Instance attribution assigns a score $\mathcal{I}(\hat{y_t}, z_i)$ to the training samples z_i that reflects a measure of importance. The scalar importance $\mathcal{I}(\hat{y_t}, z_i)$ could be derived via various methods:

- Leave-one-out retraining
- Influence functions, which is a formal approximation of the change in $\hat{y_t}$ when z_i is up-weighted
- Heuristic methods approximating influence functions

2.3.1. Leave-One-Out Retraining

With leave-one-out retraining, the effect of a singular training instance z_i on a test instance x_t is evaluated by removing that training instance from the set of all training instances such that $\hat{\mathcal{D}} = \mathcal{D} \setminus z_i$, retraining the model, and making a new prediction on the same test instance. The resulting difference in loss between the two predictions $\hat{y_t}$ and $\hat{y_t}'$ is the attributed effect of the removed training instance. Leave-one-out retraining is a naive yet inefficient approach. This method is attractive due to its ease of implementation as it can virtually be applied to any machine learning algorithm [11]. Using this method is, in fact, very straightforward and convenient if we investigate the effect of a few data points or if the model we train is simple enough to manage with a small dataset. In reality, models are too complex to train even once for a single example, and most datasets are too large to repeat this procedure for every data point.

2.3.2. Influence Functions

Despite gaining a recent surge of interest within XAI, influence functions are not a recent innovation. In fact, it is an established technique from robust statistics [19, 67] originally developed for regression models. It primarily informs how strongly the model parameters or predictions depend on a training instance. Influence functions address the impracticality of the naive leave-one-out retraining method. The idea is to approximate the leave-one-out procedure results without explicitly retraining the model from scratch. Koh and Liang [42] propose using IF as an XAI method for attributing model predictions to training instances. With this approach, the prohibitively inefficient leave-one-out retraining method can be approximated. Instead of deleting a single training instance and retraining the model to observe the change in model parameters, $\hat{\theta}_{-z} - \hat{\theta}$, this method simulates the removal of a training sample by *upweighting* the loss of the sample, (*empirical risk*), by a small ϵ in the sum of the loss over the training data. This produces the new parameters $\hat{\theta}_{\epsilon,z}$. Upweighting a sample can be considered as forcing the model to fit this particular sample harder than other training samples. For example, if a passage is highly relevant to a query in the test pairs, upweighting this sample further increases the model's confidence in predicting this query passage pair as relevant. The change in the loss on a particular test point z_{test} between the original model and the model with the upweighted instance (i.e $\mathcal{L}(x_{test}, \hat{\theta}_{\epsilon,z}) - \mathcal{L}(z_{test}, \hat{\theta})$quantifies the influence The resulting new model parameters become:

$$\hat{\theta}_{\epsilon,z} = \arg\min_{\theta \in \Theta} (1-\epsilon) \frac{1}{n} \sum_{i=1}^{n} L\left(z_i, \theta\right) + \epsilon L(z, \theta) \tag{2.11}$$

where θ is the model parameter vector, z_i the training data, $\hat{\theta}_{\epsilon,z}$ the parameter vector after up-weighting instance z by a small ϵ. The influence of a particular training instance z on a test prediction is then defined as:

$$\mathcal{I}_{up,loss}\left(z, z_{test}\right) = -\nabla_{\theta} L\left(z_{test}, \hat{\theta}\right)^{\top} H_{\hat{\theta}}^{-1} \nabla_{\theta} L(z, \hat{\theta}) \tag{2.12}$$

The main assumption is that the empirical risk is twice differentiable and strictly convex with respect to the parameter vector (θ). Because deep neural networks have millions of parameters, explicitly computing and inverting the Hessian matrix of the empirical risk is computationally too expensive. Koh and Liang [42] avoid this by using Hessian vector products(HVPs) [2, 55].

2.3.3. Heuristic Approximations of Influence Functions

Over the years, several modifications to influence functions were proposed to address the computational bottlenecks and scalability shortcomings in various applications. The two central bottlenecks are the computation of the inverse Hessian and the computation of influence values for each of the training data points. FastIF [29], proposed as an improvement to IF, uses top-k nearest neighbors of the test point to reduce the computation space of influence values from the entire training set to a subset of *promising* data points. FAISS [38] implements data structures for storing and performing fast k-selection search operations. Employing FAISS, some work cache and accelerate the k-nearest neighbors (kNN) lookup of hidden representations of input sequences to identify influential training examples.

A potential problem with using IF and the gradient product to identify examples that explain predictions is the outliers and mislabelled data points dominating the rest of the examples as they incur high loss gradients. This phenomenon can lead to identifying the same set of atypical training examples influential to many test examples. RelatIF [8] and several other methods [32] consider the influence of

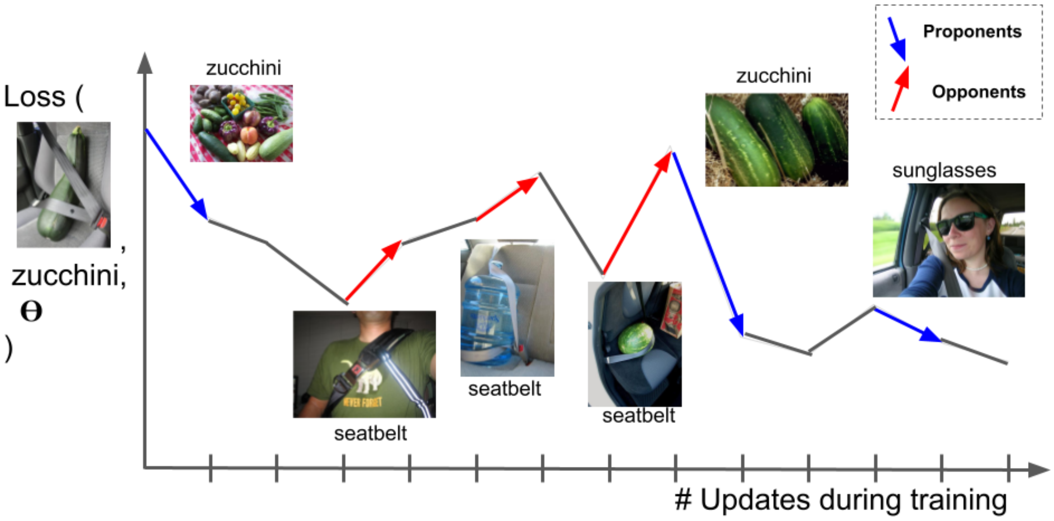


Figure 2.9: An example loss curve of the test image labeled "zucchini". Proponents are the images along the decreasing segments of the curve, and the images along the increasing trend are the opponents. Figure from [59]

a training example relative to its global effects. The main modification in these methods is substituting the dot product operation with cosine similarity, which normalizes the training gradients.

TracIn

In TracIn [59], another heuristic approximation of influence functions estimates the influence of a training example z on a test example z' as the total change in loss on z' contributed by updates from mini-batches that intuitively contain z. As iteratively repeating the training process and tracing the model parameters for each training point is not scalable, Pruthi et al. [59] use consecutive checkpoint parameter vectors to approximate the parameter vector at a specific iteration. This heuristic results in TracInCP, where there are k selected checkpoints, $w_{t_1}, w_{t_2}, ..., w_{t_k}$ are the parameters minimizing loss at iterations $t_1, t_2, ..., t_k$. Essentially, this gradient product method drops the inverse Hessian term in IF. The problem is then reduced to a dot product between the gradient of the training loss and the gradient of the loss with respect to the test example.

$$TracInCP(z, z') = \sum_{i=1}^{k} \eta_i \nabla \ell \left(w_{t_i}, z\right) \cdot \nabla \ell \left(w_{t_i}, z'\right) \tag{2.13}$$

An alternative to calling examples influential is the concept of *proponents* (excitatory instances) and *opponents* (inhibitory instances). Proponents (images along blue arrows in Figure 2.9) or "helpful" examples have positive influence values. Proponents reduce the overall training loss of a given test example. Opponents, or "harmful" examples (images along red arrows in Figure 2.9), with negative influence values increase the overall training loss.

3
Related Work

This chapter presents important research on the interpretability of neural networks, highlighting the areas of application in previous work and applications in information retrieval. A particular use case that this thesis builds upon is dataset subset selection.

3.1. Applications of Instance Attribution Methods

As all machine learning models are built using data, it would be only natural to find training instances a model deems most important. This perspective on model interpretability brings unique areas of development such as model, noisy or problematic data debugging, improving user trust in the model by showing highly influential instances, and creating adversarial examples to expose model flaws. Many beneficial use cases for analyzing the impact of individual training instances on one or a group of target predictions take the form of dataset cleaning or debugging. Biggio, Nelson, and Laskov [9] use a form of leave-one-out retraining to find adversarially created email attacks on a spam filter dataset. In contrast, Koh and Liang [42] use influence functions to generate noise that alters specific training instances to create poisoning attacks with adversarial examples. Pruthi et al. [59] propose self-influence to identify incorrectly labeled examples in the training datasets while Yeh et al. [89] use representer-points to identify erroneous data. Other work primarily uses instance attribution in a per-example manner to demonstrate its explanation capabilities. Zhou et al. [92] and Pruthi et al. [59] use different approximations of influence functions to demonstrate influential examples for individual training instances.

Another group of work focuses on finding an optimal sampling strategy to reduce the training data with instance attribution. Brunet et al. [13] use influence functions to identify subsets of documents that, when removed, reduce GloVe embedding bias the most. For linear regression models, Ting and Brochu [75] propose an optimal subsampling strategy for large datasets using influence-based methods. In text classification, an approximate influence function-based approach identifies small subsets of training examples that, if removed, can flip a prediction [86]. An alternative approach to influence functions, Paul, Ganguli, and Dziugaite [54] scores the importance of each training sample by calculating their expected loss gradient norms (GraNd score) and shows that pruning samples with small scores allow training with a significantly smaller subset of data without sacrificing much accuracy.

3.1.1. Instance Attribution in Information Retrieval Setting

There is a growing interest in instance attribution methods for XAI in various domains, including deep learning, such as image classification, object recognition, and adversarial image generation [42, 89, 59]. Additionally, it has found relevance in recommender systems [17] and NLP tasks [32, 57]. While NLP and recommender systems exhibit parallels with information retrieval (IR) tasks, especially in dealing with textual data and machine reading comprehension, instance attribution remains a relatively underexplored area in retrieval tasks. Attributing data in a text retrieval context can provide valuable insights into the query and passage representations in complex transformer models. Prior work has not investigated the training data responsible for the model to rank textual data in a particular relevance order. Limited work has been done in information retrieval to explain dense retrieval models and debug datasets using instance attribution. The application of instance attribution methods for pruning large datasets in information retrieval tasks is even less explored. These considerations have driven the motivation behind this thesis, which aims to investigate the use of instance attribution methods for passage retrieval tasks and conduct experiments on the potential application of training data subset selection.

4 Pruning Datasets With Instance Attribution

This chapter presents an overview of the experimental configuration for addressing Research Questions 1 and 2 (RQ1 and RQ2). Subsequently, a detailed description of the methodology employed within this experimental setup is presented. The thesis follows a structured format where each chapter addresses one or more research questions. The core contribution of this work is the evaluation of the proposed approach that uses influence values derived from the instance attribution method TracInCP to select the most important training examples and subsequently create more efficient training data subsets. We adopt the Tracin method to compute influence scores. To ascertain the rationale and viability of this approach, we address RQ1 and RQ2:

- **RQ1:** “How effectively can instance attribution methods help prune large datasets used in text retrieval models without significant sacrifices in performance?”
- **RQ2:** “To what extent does the proposed method of using influence scores for dataset pruning generalize to unseen data?”

4.1. Methodology

This section presents a comprehensive overview of the methodology for addressing RQ1 and RQ2. To effectively accomplish this objective, the following key components and methods are necessary:

- **Sizable Preprocessed Dataset:** The foundation of this investigation rests upon a substantial dataset preprocessed to facilitate the exploration of instance attribution within retrieval tasks, as well as the repercussions of dataset pruning. This dataset simultaneously serves as an established benchmark within the task domain, ensuring the reliability and consistency of the derived findings.
- **Task-Specific Ranking Model:** Central to this endeavor is a ranking model tailored to the task of passage re-ranking. The selected model has a relatively compact size and a moderate training duration, aligning with the study’s goals.
- **Efficient Instance Attribution Method:** An efficient instance attribution technique has been selected to pinpoint influential examples within the dataset. This method is pivotal in assessing training instance significance and subset construction.
- **Optimal Subset Selection Approach:** A set of strategic approaches have been devised and compared for selecting subsets of training data. The ideal approach should balance expected computational efficiency during training and the quality of the model predictions during testing. The goal is to ensure that model capabilities are not significantly compromised while achieving computational feasibility.

4.1.1. MSMARCO Passage Dataset Preprocessing

The MSMARCO passage dataset contains various splits [1]for training, fine-tuning, and evaluation purposes. Among those we select:

- `msmarco-passage/train` split as the main training data containing the document and query corpus
- `msmarco-passage/train/triples-v2` split for augmenting the train data to with negative examples
- `msmarco-passage/dev/small` split as the validation set for computing influence scores for each example in training data
- `msmarco-passage/trec-dl-2019` and `msmarco-passage/trec-dl-2020` splits as test sets for determining the generalization capacity of our proposed pruning method

We augment the training data with *hard negative* examples as described above. An example in each dataset split consists of a query ID, a passage ID, and a relevance score. We call an example a negative example if the relevance label of the example is annotated 0 in the dataset, conversely positive if the label is 1. The MSMARCO dataset has a sparse annotation, meaning that given a query ID, label information provided in the dataset is exclusively positive examples; thus, the assumption is that the complementary set construes the negative examples by default. Furthermore, one passage is often annotated as relevant per query; however, the `msmarco-passage/train corpus` is quite large, with nearly 8.8M passages, and contains many and near duplicates. In a real-life scenario, ranking, and retrieval of relevant passages require distinguishing relevant and non-relevant passages, necessitating retrieval models to be trained with positive and negative examples. It has recently been shown that models trained on MSMARCO retrieve a better result than the labeled ground truth answer for roughly 60% of queries [5]. Considering this phenomenon, incorporating hard negatives rather than random negatives is imperative. Concerning a specific query ID, a hard negative refers to the same query ID -passage combinations that are meaningfully non-relevant. There are various methods for extracting negative examples that are not entirely random, the simplest being BM25 negatives and more advanced mining negatives with a cross encoder [58] and using denoising [61].

While we acknowledge the impact of batch quality for this task, we use BM25 negatives as an alternative to random negatives as more advanced methods are outside the scope of this research focus. Using `msmarco-passage/train/triples-v2` split, for each positive example in the `msmarco-passage/train dataset`, we access 3 BM25 negatives with matching query ids but different passages.

The first step of generating explanations and influence scores for training instances with TracIn is training a model using the entire training set. We employ a cross-encoder (Section 2.1.5.3) model architecture, considering its capacity to handle longer input sequences like passages and high performance in inferring contextual similarity from joint input representations. We select MiniLMv2 (Section 2.1.5.1) as the base model and AutoModelSequenceClassification model from the transformers library [2] for fine-tuning for the task. The pooled MiniLM outputs act as hidden states passed as inputs to the linear layer of the classification head to produce a final relevance score output. The weights of the linear layer of the classification head are randomly initialized, and the classifier is optimized using sigmoid binary cross entropy loss function for binary relevance prediction. The binary cross entropy loss is defined in Eq. 4.1, where y are labels, and x are predictions, both in the range of $[0, 1]$. We use the optax function `optax.sigmoid_binary_cross_entropy(logits, labels)`.

$$\mathcal{L}_{BCE} = -\sum_i [y_i \log(x_i + (1 - y_i \log(1 - x_i)] \tag{4.1}$$

The loss is calculated on the logits instead of the sigmoid outputs for numerical stability. We use a batch size of 64 for training and a batch size of 3000 for evaluation. The learning rate is set to $1e-5$-throughout this work. The model is trained for five epochs, and following the TracIn method guidelines, we checkpoint three model states with the highest reduction in training loss. Table 4.1 shows the complete training and model parameter setup list.

[1] The list of all splits can be found at `https://ir-datasets.com/msmarco-passage.html`

[2] Further details on the transformers library can be found at `https://huggingface.co/docs/transformers/model_doc/auto`

Parameter	Value	Description
Model	CrossEncoder	Initialized with random weights
Base model	MiniLM-L12-H384-uncased	Pretrained model for generating sequence embeddings
Loss	Sigmoid binary cross entropy	Training objective to minimize
Hidden layers	4	Number of hidden layers
Attention heads	12	Number of attention heads
Tokenizer	MiniLM-L4-H384, vocab size =30522	Tokenizer configuration
Max input length	256	Input sequences are truncated to a maximum length of 256
Training batch size	64	
Epochs	5	Number of training epochs
Optimizer	AdamW	Optimization method
Learning rate	1e-5	AdamW hyperparameter
Evaluation interval	8000 steps	Training loss and validation metric evaluation every 8000 steps
Evaluation batch size	3000	
Checkpoint criteria	Highest reduction in training loss	Checkpoint selection criteria for TracIn

Table 4.1: The overview of hyperparameters and setup of the cross encoder model used in the implementation

4.1.2. Computing Influence Scores with TracIn

The computational time and cost of influence score calculation grow substantially for larger datasets. These scores offer a mechanism to quantitatively express how individual training examples contribute to adjusting a network's parameters throughout the training process. By assessing the change in loss gradients attributed to the exclusion of a specific training instance, influence scores offer valuable insights into the importance of each example within the training dataset. There are various approaches within instance attribution to calculate the influence of training examples. Among those, we use the TracIn approach described in section 2.3.3.1. We use the three checkpoints with the highest reduction in training loss. Despite the TracIn method presenting an approximation to the iterative training of a complete model for each training instance, the computation still requires multiple forward passes of the entire model parameters to quantify the gradient change for every text example within the validation set. Given the considerable sizes of both the training and validation sets, this operation is not scalable. In light of this, we resort to a further approximation.

The validation set contains 6.7M `scoreddocs`. This translates to an average of around 1000 query-document pairs for the 6,980 queries. Within this context, it is noteworthy that only 7,347 pairs (corresponding to the number of `qrels`) possess a label of 1, signifying relevance, and of these, 6,980 pairs have a unique query. Consequently, we only select one relevant document for each query instead of approximately 1,000. This heuristic approach primarily evaluates the significance of training examples in the context of exclusively positive relevance. This might appear counterintuitive, considering our overarching argument in section x for initially including negative examples within the training set. We find ourselves assessing the individual contributions of training data toward predicting positively labeled instances, although a significant portion of the training data carries negative labels. Furthermore, this approach does not accurately mirror the actual nature of the passage ranking task, where the objective is to discern the relevance of both relevant and non-relevant passages. To alleviate this, we apply the subsequent selection mechanisms exclusively on the positive examples and add the corresponding negative examples for each positive example to the subsets. By doing so, the pruning ratio remains unchanged, but only positive examples determine the selection of subsets.

4.1.3. Subset Selection Using Influence Scores

In the context of our experimental setup, our approach generates a 2D array of influence scores. The rows correspond to the number of queries in the validation set, while the columns represent the number of training examples, making up a $6980x1.6M$ array. These influence scores are inherently computed per example, implying that each training example within the training set is assigned a score for every query-document pair in the validation set.

Our primary objective is to identify significant training examples representing the entire test set instead of isolated pair-wise instances, as seen in most previous work. A training example might exhibit a notably high influence score for a specific query-document pair yet concurrently be scored substantially lower for another example within the validation set. The process of determining an ideal subset for selection poses a non-trivial challenge. This involves strategically choosing training examples that collectively encapsulate the importance of predicting outcomes for examples within the validation set. Because of this, it is necessary to explore various methodologies for selecting pertinent training instances. Consequently, our experimental framework includes investigating different techniques for this selection process.

Naive Average Influence Top-k and Lowest-k

The initial attempt at generating influence scores encompassing all query-document pairs within the validation is the straightforward process of averaging influence scores across rows. Subsequently, the top-k and lowest-k of positive instances are selected, with the value of k determined by the desired pruning percentage. While this approach is a simple method for producing a representative influence score set, it is, at best, naive. One potential concern is that the resulting scores for different training instances might closely converge by simply averaging influence scores. Selection based on average scores could also favor training instances with moderate scores across most query-document pairs. This selection bias could favor such instances over those that might be exceptionally important for only a limited subset of query-document pairs, even if their scores are subpar on other pairs. Given the potential validity of our concerns, we investigate alternative selection strategies that guarantee a certain amount of influential examples for each query-document pair.

Maximizing Scores for Budget

In this proposed strategy outlined in Algorithm 1, a budget is determined by the specified pruning ratio. Iteratively, the training instance with the highest influence score is selected for each query-document pair until the budget is filled. This selection guarantees the presence of at least one influential training example for every pair. Although statistically unlikely due to many training instances (1.6 million), the same training instance could potentially possess the maximum influence score for different query-document pairs within the validation set. This is possible if two training instances have similar query and document contexts or structures. To mitigate this scenario, the subsequent best training instance is chosen as the iteration progresses. The resulting subset from this method ensures that selected instances include significant training instances for each validation set example. Nonetheless, a limitation of this approach is the inability to predict in advance whether the chosen instances collectively constitute a favorable set for all validation examples.

Algorithm 1 Selection of Instances with a Budget

```
Require: Budget $B$, $S$ a $m$ x $n$ influence score matrix
  selected_indices = [ ]
  while $B > 0$ do
    for each row in $S$ do
      if $B > 0$ then
        max_index←FindIndexOfMaxElement(row)
        selected_indices⌒max_index
        $B--$
        $S[:, \text{max\_index}] = -\inf$
      end if
    end for
  end while
  return selected_indices
```

Optimization-Based

Acknowledging the limitations in the approaches above, we explore an optimization-based algorithm influenced by a methodology proposed in [87]. Here, we devise an objective function that accounts for both the influence scores and the targeted cardinality of the training data subset. This algorithm yields an optimal training data subset represented as a binary vector with values 0 and 1. Subsequently, instances associated with a binary value of 1 are retained within the subset, while instances with 0 are pruned.

We leverage CVXPY [26, 3], a Python-embedded modeling language tailored for convex optimization problems. CVXPY automatically transforms the problem into a standard form and supports multiple commercial solvers. The problem formulation for our purposes leads to a mixed-integer problem, necessitating an appropriate solver. We experimented with several solvers, including Gurobi [31] and CPLEX. Regrettably, the scale of our dataset considerably increases the number of variables in the problem, and it was impossible to obtain a solution within a reasonable timeframe using these solvers. Nonetheless, this methodology could be viable for implementations involving significantly smaller training datasets, up to a few thousand instances.

Algorithm 2 Cardinality Guaranteed Pruning with Influence Scores

```
Require: Dataset $D = z_1, ..., z_n$
Require: size of the subset, $m$
Require: influence scores, $S$
  Initialize $W \in \{0,1\}^n$
  Solve the following problem to get W:
```

$$\underset{W}{\text{maximize}} \ \|W^T S\|_2 \tag{4.2}$$

$$\text{subject to} \ \sum_{i=1}^{n} W_i = m \tag{4.3}$$

$$W \in \{0,1\}^n$$

```
  Construct the cardinality guaranteed subset $\hat{D} = \{z_i | \forall z_i \in D, W_i = 1\}$
  return Pruned dataset $\hat{D}$
```

4.1.4. Evaluation on Test Datasets

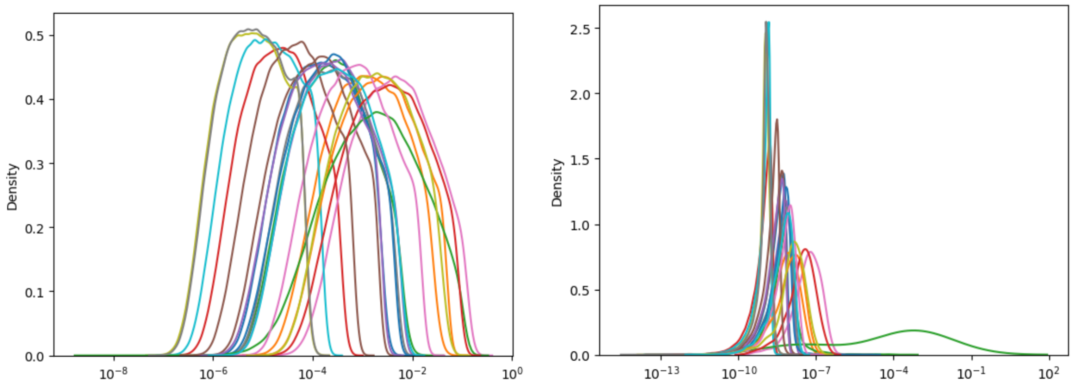
We use the `msmarco-passage/dev/trec-dl-2019/judged` and `msmarco-passage/trec-dl-2020/judged` splits as the unseen test sets for evaluating the performance of the baseline and the suggested methods in the previous chapter. Both dataset splits contain a relatively limited number of queries and corresponding relevance judgments, comprising 43 queries with 9260 relevance judgments for the first split and 54 queries with 11386 relevance judgments for the second split. Notably, the size of these evaluation sets has remained the same.

Distinct from the training and validation sets (`msmarco-passage/train` and `msmarco-passage/dev/small`), these evaluation sets present a difference in relevance judgment composition. Specifically, the relevance judgments encompass a spectrum of values from 0 to 3, indicating varied degrees of relevance. To standardize this for our analysis, we establish a lower threshold wherein pairs achieving a score of 1 and above are deemed relevant. Any judgments below this threshold are assigned a value of 0.

In the preceding section, we dedicated our efforts to training distinct CrossEncoder models tailored to specific subset ratios and pruning methodologies. To investigate generalization capabilities, we use checkpoints of these models at the step where the highest validation metrics were achieved. The pivotal factor dictating the selection of checkpointed models for each configuration rests upon the RR@10 validation performance, particularly for each model.

4.2. Results

After introducing our subset-creating methodology for the training data, we move on to the experimental phase to address our research questions (RQs). This section addresses RQ1 by presenting metric results on the validation dataset. Subsequently, we tackle RQ2 by presenting metric results on the test datasets. We use the described selection methods to analyze the model quality when trained with smaller subsets.



(a)

(b)

Figure 4.1: (Absolute) Influence score distributions of positive (a) and negative examples (b) for the randomly selected 20 validation examples. Each curve represents the influence score distribution over the entire training data for a sampled validation qrel. The influence value distributions for the positive qrels exhibit a similar pattern, while the distributions for the negative qrels are significantly different.

We train CrossEncoders using both baseline dataset subsets and dataset subsets proposed by our approach. Before comparing the models, we delve into TracIn to determine whether the assigned influence scores to training instances effectively reflect their significance for the validation set.

4.2.1. Exploratory Analysis of TracIn Influence Scores

The initial phase of our subset selection approach involves the computation of TracIn influence scores for every training sample. This method calculates influence scores individually for each of the 6,980 examples present within our constructed validation set. First and foremost, our investigation aims to validate the accuracy of our expectations concerning the distributions of influence scores. One initial concern revolved around averaging influence scores across all validation examples, potentially leading to these scores losing their interpretability due to convergence towards similar values when averaged. In Figure 4.1, we select 20 random validation examples and visualize the distribution of the influence scores of the positive examples and absolute influence score distributions for the negative examples. The distributions of scores attributed to training examples exhibit variability across distinct validation examples, more evidently in negative examples. This observation implies that while the distribution's width remains relatively consistent for positive examples, the assigned scores can vary in magnitude, contingent upon the specific validation example for which the influences are calculated.

We then investigate averaged influence scores over the validation examples in Figure 4.2. The foundational width of the violin plot for negative examples indicates that a significant proportion of the averaged influence scores for negative instances are nearer to 0 and in the negative range. For positive examples, the widest part of the violin plot is centered around the value of 0, albeit less pronounced compared to the negative examples. In this case, the averaged values also cluster around the 0 mark.

4.2.2. Training and evaluation of Cross Encoder subset models

To investigate the effectiveness of training dataset pruning using TracIn influence values, we devised an experimental setup involving the creation of subsets of varying sizes derived from the complete training set. We use two strategies mentioned in Section 4.1.3: Naive Average Top-k and Lowest-k and Budget Score Maximization. Our interest extends to comprehending the cross-encoder model's performance dynamics across smaller and larger fractions of the entire training dataset.

For clarity, we will employ the following nomenclature for the various cross encoder models derived from distinct training data selection approaches:

1. The cross-encoder trained on the original dataset will be denoted as $\text{CrossEnc}_{\text{orig}}$
2. For the model trained on a subset representing p percentage of the dataset and chosen through random selection (which serves as our baseline), we will use the notation $\text{CrossEnc}_{\text{random\_p}}$
3. The cross-encoder model resulting from utilizing the average of the top p percentage of naive influence scores will be referred to as $\text{CrossEnc}_{\text{topavr\_p}}$, while the model resulting from the top p

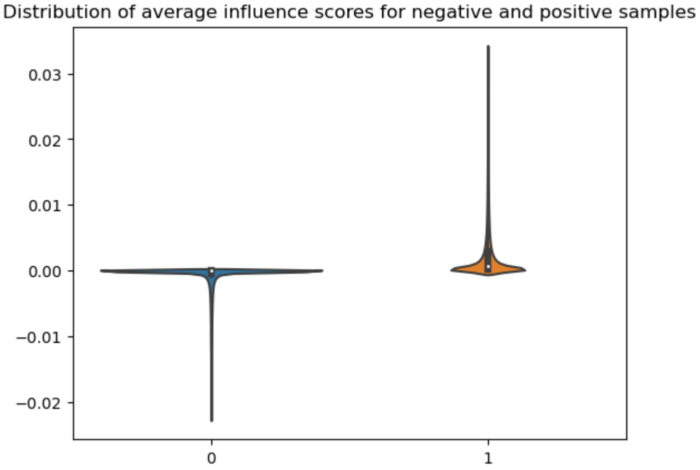


Figure 4.2: Averaged influence score distributions by label, 0 represents non-relevant query and document pairs and 1 represents relevant query and document pairs in the training data. The average influence values tend to cluster around 0. The foundational width of the violin plot for non-relevant pairs is significantly larger, suggesting majority of average influence scores for this group to be near the value 0.

percentage of the average scores will be labeled as $\text{CrossEnc}_{\text{lowestavr\_p}}$

4. Lastly, the model trained on p percentage of the training data using the budget method will be denoted as $\text{CrossEnc}_{\text{budget\_p}}$

Subset(%)	Eval(steps)	Model	RR@10	nDCG@10	selected model@step
100%	8k	$\text{CrossEnc}_{\text{orig}}$	0.326	0.383	48k
75%	8k	$\text{CrossEnc}_{\text{random\_75}}$	0.319	0.376	24k
		$\text{CrossEnc}_{\text{lowestavr\_75}}$	**0.320**	0.378	40k
		$\text{CrossEnc}_{\text{topavr\_75}}$	**0.320**	**0.379**	32k
		$\text{CrossEnc}_{\text{budget\_75}}$	0.307	0.368	32k
50%	6k	$\text{CrossEnc}_{\text{random\_50}}$	**0.314**	**0.371**	24k
		$\text{CrossEnc}_{\text{lowestavr\_50}}$	0.311	0.368	48k
		$\text{CrossEnc}_{\text{topavr\_50}}$	0.281	0.339	12k
		$\text{CrossEnc}_{\text{budget\_50}}$	0.289	0.347	12k
20%	1.5k	$\text{CrossEnc}_{\text{random\_20}}$	**0.307**	**0.364**	19.5k
		$\text{CrossEnc}_{\text{lowestavr\_20}}$	0.303	0.350	21k
		$\text{CrossEnc}_{\text{topavr\_20}}$	0.217	0.270	4.5k
		$\text{CrossEnc}_{\text{budget\_25}}$	0.210	0.262	4.5k
5%	1k	$\text{CrossEnc}_{\text{random\_5}}$	**0.276**	**0.330**	6k
		$\text{CrossEnc}_{\text{lowestavr\_5}}$	0.248	0.300	6k
		$\text{CrossEnc}_{\text{topavr\_5}}$	0.095	0.124	4k
		$\text{CrossEnc}_{\text{budget\_5}}$	0.133	0.172	5k

Table 4.2: Validation metrics of the CrossEncoder subset models, the model checkpoint step is selected based on the highest RR@ value. We find that the random subset baseline is a very strong baseline, influence based subset selection methods do not outperform any of the random baselines.

Specifically, we generated subsets equivalent to 5% , 20%, 50%, and 75% of the complete training data using the methodologies above. As a sanity check, subsets mirroring these percentages were also formed using random selection. Notably, we adopted a consistent approach employed in the subset selection procedure with influence values during this random selection process. We randomly selected only from the positive instances while forming subsets, adhering to the prescribed percentages. To ensure fair comparisons, we introduced the corresponding negative examples per randomly selected positive example externally.

Table 4.2 presents a comprehensive overview of the performance exhibited by the distinct cross-encoder models. These models were trained with the subsets constructed with the selection methods outlined earlier. The training duration for all cross-encoder models spanned five epochs while maintaining a consistent set of model hyperparameters, as detailed in Table 4.1. An adjustment was made to the evaluation frequency in light of the reduced dataset sizes in the smaller subsets. To accommodate this, smaller subsets were subjected to a correspondingly shorter evaluation interval, thereby ensuring a fair evaluation considering the reduced number of total steps.

We do not observe any interesting patterns for CrossEncoder models trained with 75 percent of the total data amount. Notably, both RR@10 and NDCG@10 metrics attain peak values for $\text{CrossEnc}_{\text{topavr\_75}}$. However, a closer examination of the NDCG and RR scores reveals that the values across all model configurations are remarkably proximate. This proximity in scores across the different methods suggests that there might not be a discernible, significant distinction among the proposed techniques compared to the random baseline at this particular percentage.

At the 50 percent data proportion juncture, the highest validation metrics for both RR@10 and NDCG@10 are obtained with the $\text{CrossEnc}_{\text{random\_50}}$ model. This outcome is intriguing, as our initial anticipation positioned the random baseline as a relatively lenient benchmark. However, it surprisingly outperforms all the proposed methodologies. Nevertheless, the distinction with the $\text{CrossEnc}_{\text{lowestavr\_50}}$ model remains modest. Interestingly, this deviation from expectation conflicts with the contexts of TracInCP research. As per the consensus, higher influence values are typically associated with proponents of validation examples, thereby reducing the overall training loss for such instances. Conversely, the inverse holds for opponents. This leads us to anticipate that the subsets constructed predominantly from proponents would exhibit notably superior performance to subsets dominated by opponents. However, this hypothesis is not corroborated by the results. In the temporal dimension, we also notice that the steps at which the models within the 50 percent group attain their highest validation metrics demonstrate a degree of variability. Specifically, $\text{CrossEnc}_{\text{topavr\_50}}$ and $\text{CrossEnc}_{\text{budget\_50}}$ reach their peak validation metrics at considerably earlier steps when compared to $\text{CrossEnc}_{\text{random\_50}}$ and $\text{CrossEnc}_{\text{lowestavr\_50}}$.

At 20 percent of the data proportion group, the highest validation metrics for RR@10 and NDCG@10 are obtained again with the $\text{CrossEnc}_{\text{random\_50}}$ model. We observe a similar pattern as we did at the 50 percent data proportion group where $\text{CrossEnc}_{\text{random\_20}}$ and $\text{CrossEnc}_{\text{lowestavr\_20}}$ exhibit better performance with closely aligned metrics. The bottom two performances are by $\text{CrossEnc}_{\text{topavr\_20}}$ and $\text{CrossEnc}_{\text{budget\_20}}$. However, this group's contrast in the highest performance step is more evident. It appears that $\text{CrossEnc}_{\text{topavr\_20}}$ and $\text{CrossEnc}_{\text{budget\_20}}$ models attain the highest validation metrics very early during the training phase, experiencing a decline in performance afterward. In contrast, the other models reach their performance peak at a relatively later phase.

Even at a mere 5 percent proportion of the total dataset, we observe commendable performance from both $\text{CrossEnc}_{\text{random\_5}}$ and $\text{CrossEnc}_{\text{lowestavr\_5}}$ models. While there is a discernible reduction in validation performance compared to the baseline model, $\text{CrossEnc}_{\text{orig}}$, this diminishment is not as severe as initially anticipated. Remarkably, the NDCG@10 scores for these two models exhibit closer proximity to the scores of the original model than the RR@10 scores. Conversely, both $\text{CrossEnc}_{\text{topavr\_5}}$ and $\text{CrossEnc}_{\text{topavr\_5}}$ demonstrate notably inferior scores for both RR@10 and NDCG@10. This significant performance discrepancy indicates that these two models struggle to maintain competitiveness at this reduced dataset proportion.

4.2.3. Evaluation of the Trained Models on Unseen Datasets

In the preceding section, we dedicated our efforts to training distinct CrossEncoder models tailored to specific subset ratios and pruning methodologies. To investigate generalization capabilities, we use checkpoints of these models at the step where the highest validation metrics were achieved. The pivotal factor dictating the selection of checkpointed models for each configuration rests upon the RR@10 validation performance, particularly for each model.

In this section, we present the retrieval performance metrics of the trained models using TracIn scores on evaluation sets. We analyze the behavior of the model quality when trained with smaller subsets of described methods.

Subset(%)	Model	RR@10	nDCG@10	model@step
		trec-dl-2019/judged		
100%	CrossEnc$_{\text{orig}}$	0.977	0.676	48k
75%	CrossEnc$_{\text{random\_75}}$	0.944	0.667	24k
	CrossEnc$_{\text{lowestavr\_75}}$	**0.958**	0.667	40k
	CrossEnc$_{\text{topavr\_75}}$	**0.958**	**0.673**	32k
	CrossEnc$_{\text{budget\_75}}$	0.936	0.664	32k
50%	CrossEnc$_{\text{random\_50}}$	0.950	**0.664**	24k
	CrossEnc$_{\text{lowestavr\_50}}$	**0.955**	0.656	48k
	CrossEnc$_{\text{topavr\_50}}$	0.928	0.626	12k
	CrossEnc$_{\text{budget\_50}}$	0.925	0.633	12k
20%	CrossEnc$_{\text{random\_20}}$	**0.939**	**0.646**	19.5k
	CrossEnc$_{\text{lowestavr\_20}}$	0.904	0.625	21k
	CrossEnc$_{\text{topavr\_20}}$	0.829	0.551	4.5k
	CrossEnc$_{\text{budget\_25}}$	0.825	0.542	4.5k
5%	CrossEnc$_{\text{random\_5}}$	**0.891**	**0.612**	6k
	CrossEnc$_{\text{lowestavr\_5}}$	0.817	0.554	6k
	CrossEnc$_{\text{topavr\_5}}$	0.657	0.348	4k
	CrossEnc$_{\text{budget\_5}}$	0.707	0.405	5k
		trec-dl-2020/judged		
100%	CrossEnc$_{\text{orig}}$	0.885	0.654	48k
75%	CrossEnc$_{\text{random\_75}}$	**0.917**	0.660	24k
	CrossEnc$_{\text{lowestavr\_75}}$	0.894	0.639	40k
	CrossEnc$_{\text{topavr\_75}}$	0.907	**0.669**	32k
	CrossEnc$_{\text{budget\_75}}$	0.913	0.649	32k
50%	CrossEnc$_{\text{random\_50}}$	**0.907**	**0.650**	24k
	CrossEnc$_{\text{lowestavr\_50}}$	0.890	0.644	48k
	CrossEnc$_{\text{topavr\_50}}$	0.873	0.625	12k
	CrossEnc$_{\text{budget\_50}}$	0.883	0.634	12k
20%	CrossEnc$_{\text{random\_20}}$	0.891	**0.640**	19.5k
	CrossEnc$_{\text{lowestavr\_20}}$	**0.910**	0.623	21k
	CrossEnc$_{\text{topavr\_20}}$	0.867	0.561	4.5k
	CrossEnc$_{\text{budget\_25}}$	0.823	0.543	4.5k
5%	CrossEnc$_{\text{random\_5}}$	0.893	**0.597**	6k
	CrossEnc$_{\text{lowestavr\_5}}$	**0.895**	0.547	6k
	CrossEnc$_{\text{topavr\_5}}$	0.544	0.286	4k
	CrossEnc$_{\text{budget\_5}}$	0.715	0.370	5k

Table 4.3: Test performance of the CrossEncoder subset models on two test sets trec-dl-2019/judged and trec-dl-2020/judged. We observe that random data subsets are a strong baseline.

We present the NDCG@10 and RR@10 metrics corresponding to the two unseen MSMARCO splits in Table 4.3. We do not identify any pattern conclusively indicating a prevalent trend of superior performance among the distinct subset percentage quantity groups for both datasets.

The earlier experiment exhibited a consistent trend, where the random subset selection method, our baseline, tended to outperform models trained with TracIn-guided dataset subsets. However, this experiment diverges from the previous pattern. The results do not align with the prior observation. It becomes apparent that the models' performance characteristics fail to replicate when the evaluation datasets differ from those from which the influence of training instances was initially inferred.

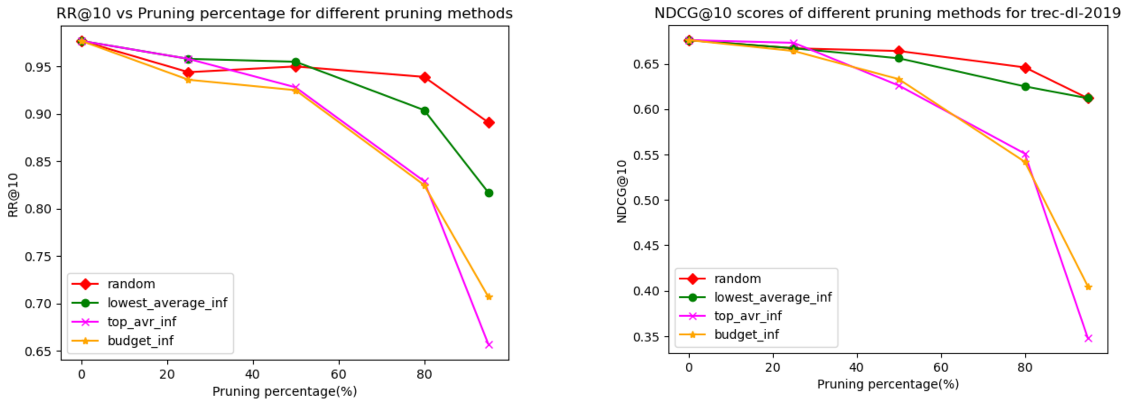


Figure 4.3: NDCG@10 and RR@10 evaluation of CrossEnc models trained different subset selection methods on trec-dl-2029/judged. The influence-based subset selection methods show higher performance when pruning percentage is small. At higher pruning percentages they perform significantly worse than random baselines.

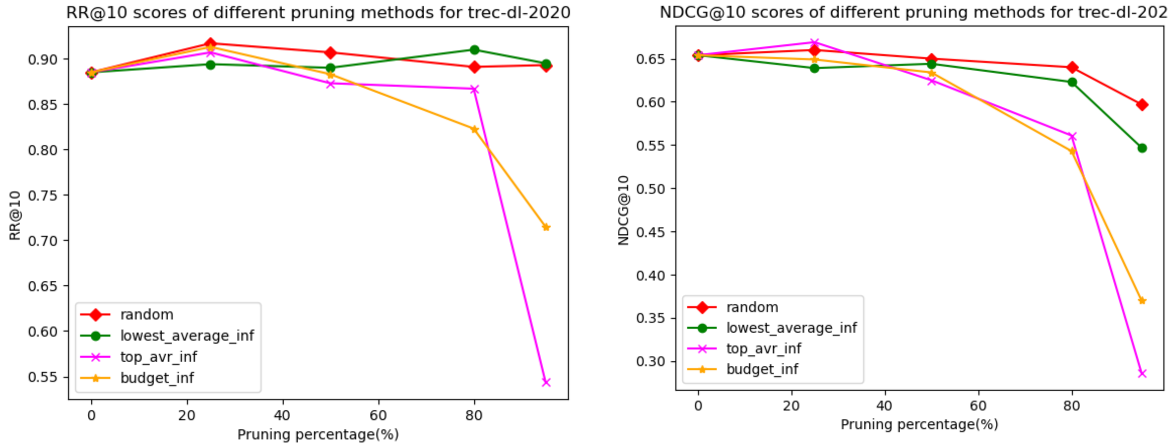


Figure 4.4: NDCG@10 and RR@10 evaluation of CrossEnc models trained different subset selection methods on trec-dl-2020/judged. Lowest average influence method displays similar performance to random baselines, top average and budget influence methods' performance decays as the pruning percentage increases.

We compare the performance decay of CrossEnc models trained in different subset configuration approaches in Figure 4.3 and Figure 4.4. For the first evaluation split, trec-dl-2019/judged, the RR@10 performance of models displays a similar decline as the amount of data used to train the models decreases. Especially when training data is pruned for more than 50%, the difference in the efficiency of the pruning methods is more apparent. Using smaller data, CrossEnc$_{\text{random\_5}}$ performs significantly higher than other pruning methods. When low amounts of the dataset are pruned, using the lowest and top influential positive examples and their negative counterparts appears to achieve slightly higher performance scores. For the NDCG@10 performance, pruning the dataset at random and using the lowest average TracIn scores appears to perform very similarly.

For the second evaluation split, trec-dl-2020/judged, we observe a very steep reduction in performance for both RR@10 and NDCG@10 metrics of CrossEnc$_{\text{budget}}$ and CrossEnc$_{\text{topavr}}$ model groups compared to the CrossEnc$_{\text{random}}$ and CrossEnc$_{\text{lowestavr}}$ model groups.

4.3. The Effect Of Randomly Selecting from a Distribution

The results obtained through our initial approach utilizing influence scores of the positive training examples do not align with our initial hypothesis. We suspect that sampling the varying percentages of a very large training dataset could produce baseline subsets with similar distributions to the original training set. In such cases, many samples drawn from this distribution could exhibit similar characteristics, potentially rendering them a more challenging baseline for comparison. In contrast, when compared to the random baseline, our method for selecting influential examples may not necessarily reflect the same underlying distribution. Therefore, we investigate whether incorporating varying percentages of randomly selected and influence-based data into subsets improves the performance.

Procedure: For each of the subset sizes, 5%, 20%, 50% and 75%, we construct the training data as follows:

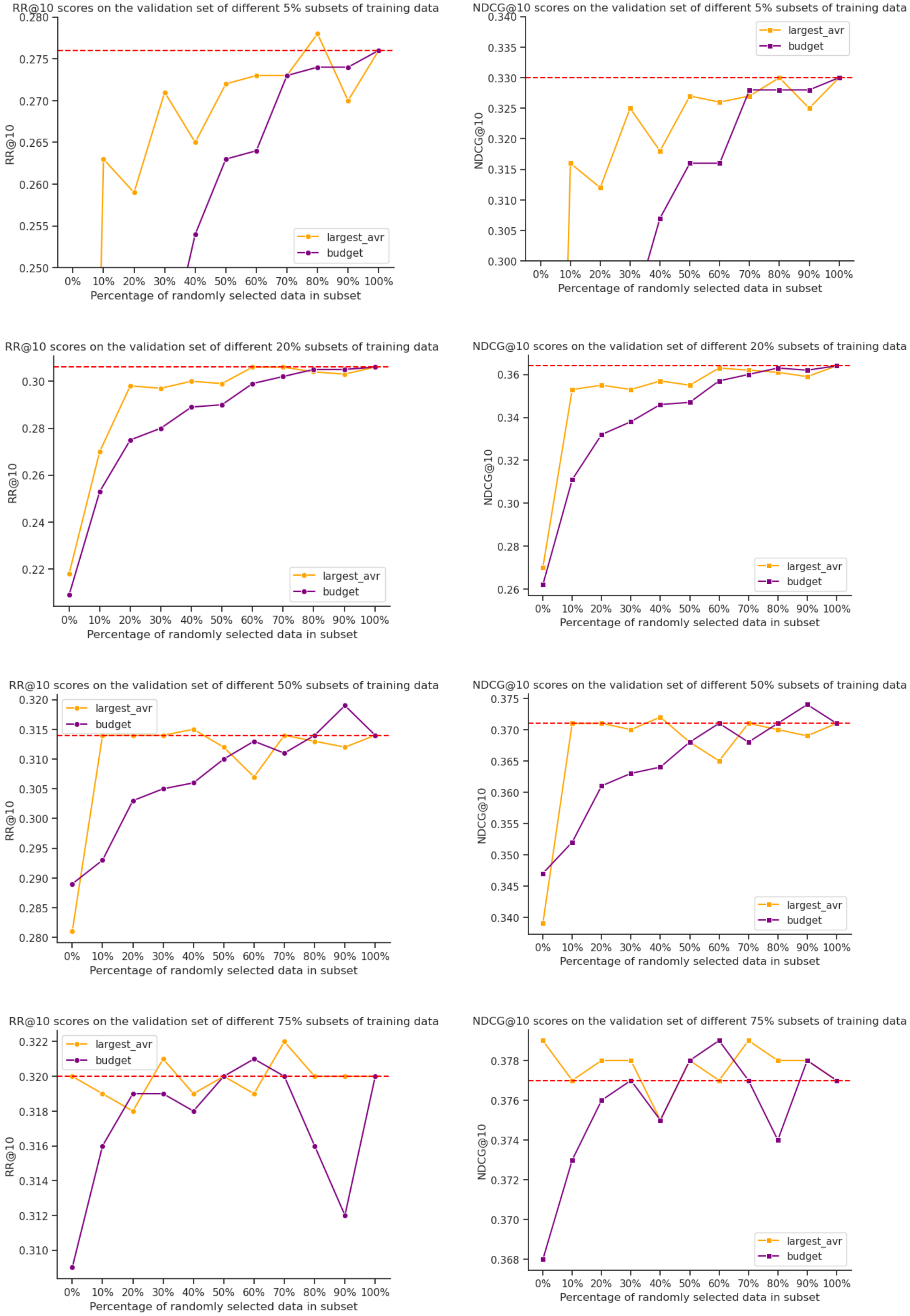
1. We construct 10 different training data variations, where the amount of data coming from random selection ranges from 0% to 100%, incrementing by 10%
2. We repeat this process for each subset size (5%, 20%, 50% and 75% of the full training data) for both average influence and budget influence subset selection approaches
3. We train a model for each training data configuration
4. For each subset size, we examine whether the reported validation metric RR@10 exceeds the random baseline scores in Figure 4.5 for the validation dataset and when which percentage of random data is included. Each subset's random selection baseline scores correspond to the red horizontal line in their respective graphs.
5. We report RR@10 and NDCG@10 metrics on trec19 and trec20 test datasets for the point where the highest validation metric was recorded. (The full results of the test metrics can be found in Appendix C.)

Subset (%)	Model config	val RR@10	val nDCG@10	trec19 RR@10	trec19 nDCG@10	trec20 RR@10	trec20 nDCG@10
100%	original model	0.326	0.383	0.977	0.676	0.885	0.654
75%	largest avr 70% random data	**0.322**	**0.379**	**0.948**	**0.669**	**0.934**	**0.671**
	baseline	0.320	0.377	0.944	0.667	0.917	0.660
50%	budget 90% random data	**0.319**	**0.374**	**0.970**	**0.668**	0.902	0.648
	baseline	0.314	0.371	0.950	0.664	0.907	0.650
20%	largest avr 60% random data	**0.306**	0.363	0.940	0.634	**0.911**	0.639
	baseline	0.306	0.364	0.940	0.650	0.908	0.645
5%	largest avr 80% random data	**0.278**	**0.330**	0.862	0.592	**0.911**	**0.614**
	baseline	0.276	0.330	0.892	0.612	0.893	0.597

Table 4.4: Comparison of results between the baseline models and new subset configurations that incorporate different random and influence sampling data rates. Except for the 50% subset level, the largest average method combined with random data displays the highest validation performance. Combining random and influence-based data for subset selection improves the validation performance over baselines by a small margin. There is no significant improvement over test performance.

The results summarized in Table 4.4 do not definitively confirm an improvement in model performance when varying percentages of randomly selected and influence-based data into subsets are combined. Except for the 50% subset level, the largest average method combined with random data displays the highest validation performance. Combining random and influence-based data for subset selection improves the validation performance over baselines by a small margin. There is no significant improvement over test performance.

Figure 4.5: RR@10 and NDCG@10 metrics on the validation dataset for subsets 5%, 20%, 50% and 75% constructed with different ratios of random and influence-based dataset sampling



5

Patterns in Pruned Examples

In this chapter, our primary objective is to identify patterns within the pruned examples and, in doing so, gain insights into why the outcomes of the previous experiments in Chapter 4 did not align with our initial hypotheses. Specifically, we aim to address the following research question and sub-question:

- **RQ3:** What patterns can be identified in the examples pruned using instance attribution?
- **RQ3.1:** To what extent can instance attribution techniques justify the influence of selected examples in a text retrieval model?

5.1. Qualitative Analysis of Influential Examples

We analyze the most influential examples for various query document pairs in the validation set. This analysis serves as a sanity check for validating our method for inferring influence functions as anticipated. Our experiments differ from most prior applications of TracIn and other instance attribution methods as they rely solely on textual data, whereas previous applications often involve image data. Given the nuanced context of text data, validating the results at a glance is naturally more challenging.

In our analysis, we introduce a query-document pair with a ground truth relevance label of 1, indicating that it is indeed a relevant pair. This pair is selected from the validation set from which the influence scores were computed. Our hypothesis for this sanity check is as follows: for a test instance belonging to class "r", the most influential examples, referred to as "proponents", should consist of training instances that closely resemble the test pair or support the prediction of the test pair as relevant and share the same relevance label "r". Again, in contrast to assessing images, defining what constitutes a resemblance in the context of textual data is notably more intricate and nuanced. Because of this, we elaborate on additional potential factors that explain why a training example is considered influential even when its similarity may not be readily apparent at first glance.

Procedure: The procedure itself does not involve any further training or inference as we have computed the influence values of all training instances for each validation example in earlier experiments. Using the previous influence matrix, we obtain the top 5 influential examples (proponents) of randomly selected test instances in the validation set. This way, we pinpoint which specific training instances played a decisive role in shaping the prediction of a given validation item.

pred: 0.612 **ground truth:**1	**validation query:**.......is considered the father of modern medicine.	**validation passage:** TRUE. Hippocrates is considered the father of modern medicine because he did not believe ...
relevance	**query text**	**passage text**
1	.... is the color of the visible spectrum with the longest wavelength.	The color red is the longest wavelength in the visible light spectrum because the less energy a wave carries the longer the wavelength and red carries the least energy of all … the colors in the visible spectrum.
1	..... is the law that prevents american companies from bribing foreign officials.	President Trump reportedly vented to Secretary of State Rex Tillerson about federal laws preventing ...
1	....., the leading philosopher of the twentieth century, made significant contributions as a playwright, novelist, journalist,	Sartre's activity as a playwright, novelist and literary critic gave his ideas extraordinary reach; his novel ...
1	...... are tradition bound, suspicious of changes and adopt the innovation only when it has become something of a tradition itself	The late majority are skeptical—they adopt an innovation only after a majority of people have tried it....
1	..... the highest point of elevation in australia is located in the australian alps	The tallest mountain in Australia is Mt Kosciuszko, at 2228m (some sources say 2229m), or 7310 feet. ...

Table 5.1: Top 5 influential training examples for validation query id: 9083, passage id: 7067273 example in qrels set

Tables 5.2 and 5.1 visualize the results of TracInCP proponents. Table 5.1 supports our hypothesis that the proponents would be similar examples to the validation query. However, the proponents on 5.2 are not as significantly similar to the validation example. A closer examination of these figures reveals that the opponents tend to be examples that support the prediction of the given validation set example. Specifically, in Table 5.2, the passage presented serves as an answer for the query but is also a continuation of the exact query phrase. The top 5 proponents for this example exhibit minimal contextual similarity to this instance. However, their relevance determination strategy appears to be similar: the validation query lacks a clear and direct question or information need (i.e., it lacks question words like “what”, “when” etc., or a question-like sentence structure), yet the retrieved passage contains the answer, effectively continuing from where the query was truncated. The passages of the proponents similarly contain answers to queries. Our understanding is that these opponents may either involve challenging queries (e.g., questions that encompass multiple inquiries in one like “What is artificial selection and give another name for it?”, or poorly constructed queries like “Do batteries in a travel... have to be working for the propane fridge to work”, or passages that indirectly address the queries or present the answer towards the conclusion. For instance, consider the passage mentioning “Russian Black Bread”, which touches upon several ingredients' impact on the loaf's color and flavor.

Conversely, in Table 5.1, we can quickly identify a pattern where the structure of the proponent queries closely mirrors the query structure of the given example. All proponents consistently commence with a “...” segment, indicating the search intent is focused on identifying a word that matches the description in the remainder of the query phrase. Likewise, all retrieved passages contain the answer and matching keywords that align with the queries.

pred: 0.994 **ground truth:**1	**validation query:** when delivering a briefing, confidence, enthusiasm, and body language are classified under	**validation passage:** When delivering a briefing confidence enthusiasm and body language are classified under nonverbal consideration.
relevance	**query text**	**passage text**
1	what is artificial selection and give another name for it?	A new life may be born out of artificial selection or natural selection. This article will provide you with answers on why organisms have different traits...
1	what do fennel seeds taste like?	Russian Black Bread...Cocoa and coffee powders darken the loaf, and caraway and fennel seeds impart just a bit of licorice flavor...
1	do the batteries in a travel have to be working for the elect./propane fridge to work	Wilderness Trailer's fridge will work on propane but not on battery...
1	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New-Used Cars,...
1	accounting when does a transaction hit the books?	... 'Trade Date Accounting'. A method company accountants and bookkeepers use to record transactions that take place on the date at which an agreement has been entered (the trade date), and not on the date the transaction has been finalized (the settlement date).

Table 5.2: Top 5 influential training examples (proponents) for validation qrel(query id: 1007382, passage id: 7251891)

5.1.1. Analysis of the correctly predicted and mispredicted examples

Up to this point, we have established a general understanding that the influential examples, or proponents, for the test instances within the validation set align with logical expectations upon human inspection. In this section, our focus is on understanding the characteristics of opponents and proponents across varying prediction confidence levels. Given that the test instances within the validation set are exclusively positive examples, we stratify examples in the following manner:

- **Correct Predictions:** For this category, we randomly select a few examples from the qrels that received high predicted scores, typically within the top 50 values. These high scores typically range from approximately 0.999 to 0.995. These instances represent cases where the model's relevance predictions are correct.
- **Incorrect Predictions:** In this scenario, the predictions are incorrect, with the predicted probability falling below a designated threshold, near 0. We randomly select a few examples from the qrels where the predicted probabilities are at the bottom 50. These low probabilities represent instances where the model's predictions do not align with the relevance labels.

Building upon previous research, we hypothesize that when proponents for a given prediction point are removed (i.e., $\hat{y}'$), we anticipate a decrease in the predicted value for the ground truth class. Conversely, if we were to remove an opponent, we would expect the predicted value to increase. We identify two viable approaches to explore this hypothesis: Building upon previous research, we hypothesize that when proponents for a given prediction point are removed (i.e., $\hat{y}'$), we anticipate a decrease in the predicted value for the ground truth class. Conversely, if we were to remove an opponent, we would expect the predicted value to increase. We identify two viable approaches to explore this hypothesis:

- **Training the Model Anew:** This method entails retraining the model from scratch and comparing prediction probabilities before and after removing specific points, known as leave-one-out retraining. This approach is, in fact, similar to our experiments in RQ1 and RQ2. However, in previous experiments, we primarily concentrated on removing entire groups of points, as opposed to the current investigation, where we are specifically exploring the effects of removing individual points.
- **Human Interpretation:** In our approach, we opt for this method. Instead of retraining the model to compare the predicted probabilities of the same examples before and after proponent and opponent removal, we use human judgment to analyze the relationship between samples from the validation set and their identified proponents and opponents.

	pred: 0.996 **ground truth:**1	**validation query:**how many calories in one fried oyster	**validation passage:** The calories in Fried Oyster per 29.1g(1 roll) is 57 calories. Fried Oyster is calculated to be 196Cal per 100 grams making 80Cal equivalent to 40.82g with 3.74g of mostly carbohydrates ...
	relevance	**query text**	**passage text**
proponents	1	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake,....
	1	developmental psychiatrist make yearly	An average yearly salary for a psychiatrist ranges from between $160,000 to $230,000. ...
	1	marcoa publishing dallas tx phone number	Dallas TX 75244 : No searches yet (972) 386-8861: 9723868861: 3 Listings found ...
opponents	0	how long do bernese mountain dogs live	How big do bernese mountain dogs get? Bernese mountain dogs get pretty big,the males can get to 90-120 pounds...
	0	turkey sausage calories	In Rumbamel's Cabbage and Turkey Sausage ... Calories: 219, Fat: 8g, Carbs: 20g, Protein: 16g, ...
	0	how long to bake a pork chop in the oven...	Preheat oven to 375°F. 2 Place pork filet in shallow roasting pan. 3 Cover with foil and bake for 40 minutes...

Table 5.3: Top 3 opponents and proponents of a correctly predicted validation query document pair with high confidence

The validation sample in Table 5.3 is an example of a strong prediction made by the model. The predicted label 0.996 is very close to the ground truth label of relevance, 1. The query of the validation sample contains the terms “calories” and oyster, which are closely related, particularly within the context of food. The first proponent contains “calories” in the retrieved passage. Although the query may not explicitly be about food, the terms “carbs” and “sugar” are highly relevant to food and calories. This proponent supports the model’s prediction, and removing it might decrease the predicted label probability. The remaining proponents, while not related to nutrition, both contain numbers. One is in the form of a phone number, and the other provides salary figures. Notably, numerical terms regarding calories dominate the retrieved relevant passage of the validation example. It appears that similarity in the structure of the passages (in this case, containing numbers) also guides the model predictions.

Similar to the general pattern observed so far, the top opponents are predominantly non-relevant query passage pairs in the training data, as the influence values of those for a relevant validation pair are always negative. The first opponent in Table 5.3 contains a query about “Bernese” mountain dogs. This opponent is interesting because the closely similar word *bernaise* is a sauce in the culinary world. This opponent suggests the model is sensitive to exact or near-exact term matches. The query was likely evaluated wrongly as food-related, decreasing the model’s confidence in its prediction. The other two opponents are associated with food and nutrition, specifically “turkey sausage” and “pork chop”. One even has an exact word match, the term “calories” in the query. Interestingly, these examples are mislabeled in the dataset. Human observation suggests that the retrieved passages are relevant. These examples are apparent opponents, as their queries are semantically similar to the validation query, but their relevance label is annotated as non-relevant. This suggests the model likely identifies similar terms and context and samples containing similar queries, but a non-supporting relevance label confuses the model.

	pred: 0.001 < **ground truth:**1	**validation query:**what is the prize money for women on the eu ski	**validation passage:** The prize money for the tournament will be a record high of £1,800,000 in total. The winner’s prize money has increased from £350,000 to £400,000.
	relevance	**query text**	**passage text**
proponents	1	can board members get a salary in non-profit calgary	The average salary for non-profit board member jobs is $68,000. Average nonprofit board member salaries can vary greatly due to company, location, industry, experience and benefits. This salary was calculated using ...
	1	how much are hollow scream tickets in Virginia	Busch Gardens is offering single-day admission tickets valid during its Howl-O-Scream event for $45 through Groupon...
	1	cost of owning a pool oklahoma	First, let’s look at the cost of actually building a pool. Obviously, the fancier the pool, the more expensive it will be. On average, an in-ground pool can cost anything from $12,000 to $50,000....

opponents	0	where is the apple store in the mall in greenville	How to get here: The Apple Store is located in Westfield Garden State Plaza, in the mall center opposite JC Penney. Westfield Garden State Plaza is located at the intersection of Garden State Parkway/Route 4 and Route 17 in southern Paramus....
	0	pa department of revenue contact	Massachusetts Department of Revenue (DOR) Phone Number For Customer Service... to contact customer service of Massachusetts to get official support for solving technical problems and helpline is 800-392-6089 for customer support...
	0	is the louvre open on bastille day	The Louvre is open evenings until 9.45pm on Wednesdays and Fridays. Tickets for the permanent exhibitions is 8.50 euros before 6pm and 6 euros after 6pm...

Table 5.4: Top 3 opponents and proponents of a mispredicted validation query document pair

5.2. Contextual Patterns in Pruned Examples

In this section, we are trying to answer the question “Is there any observable pattern in the training data that displays low influence scores?”. Before elaborating on potential patterns among pruned examples in our previous experiments, it is imperative to note that the removed examples were determined based on a representative relevant query-document pair, with the non-relevant pairs either pruned alongside the relevant ones or retained together. Focusing on the relevant query-document pairs within the training set, we aggregate the top 10 lowest-scoring example occurrences for each prediction point corresponding to the 6,980 query relevance labels (qrels) within the validation set.

Topic cluster	Description of query intent or topic within cluster
PN	Phone number inquiry of a facility, person, company of service
LD	Location, direction, or general information inquiry of an entity, often facility, geographic region or company
T	Trivia questions about entertainment, concepts and arts
D	Definition or meaning of a concept or term, often scientific or domain-specific

Table 5.5: Topic or query intent clustering scheme

We identify 18 positive instances that consistently rank within the bottom ten influence scores compared to the remaining 1.6 million training instances. These 18 instances maintain this low ranking for at least 10 percent of all qrels in the validation set, corresponding to an overlapping range from 779 to 6,609 occurrences across 6,980 qrels. This observation bears significance, considering the substantial training data these 18 instances contend with. Their consistent placement within the bottom ten influence scores across more than one-tenth of all test points warrants further investigation into the underlying factors contributing to this pattern. We cluster these instances based on the query topic or intent, where the topic and intent of queries are not necessarily exclusive. Upon scrutinizing the content of the queries, our analysis has unveiled four predominant query groups, which are concisely summarized in Table 5.5. Any query that does not align with these predefined groups is labeled “Other”.

	PN	LD	T	D	Other	Total
Most frequent low-influence examples (for >=10% of qrels)	50.0%	33.3%	11.1%		5.6%	18
Most frequent low-influence examples (for >=5% of qrels)	34.0%	24.0%	10.0%	22.0%	10.0%	50

Table 5.6: Query topic distributions across the training examples scoring lowest influence for qrels in validation set (for at least 10% and 5% of all qrels (6,980 total)

Table 5.6 presents the incidence ratios of the training example queries categorized by topic. These queries specifically belong to the training examples that achieve bottom 10 influence scores for at least 10 and 5 percent of all qrels in the validation set, respectively. In context, the training examples that exhibit the lowest influence scores for a minimum of 10 percent of the validation set are notable. The most frequently occurring example in this category appears 6,609 times, while the eighteenth most frequent example is observed in 779 instances across 6,980 examples in the dataset. This observation underscores the significance of these instances and their consistent impact on a substantial portion of the validation data. Among these training instances, approximately half of the queries fall under the PN cluster, all of which pertain to inquiries about phone numbers. Queries related to the LD topic cluster constitute the second most significant group, comprising about one-third of the total.

Examining the second row of the table allows us to put this distribution into perspective, focusing on the first 50 most frequent queries grouped by topic. This reaffirms the dominance of the PN and LD groups. However, we also observe the emergence of the D group, where the queries revolve around seeking definitions of concepts or domain-specific terms that are not commonly encountered in everyday life.

An interesting thing to note is that all of these queries are extracted from the query-passage pairs that have the ground truth label as relevant. Examining the pair dynamics, we also confirm that the pairs are correctly labeled in the train set. Nearly all the corresponding passages contain either the exact or very comprehensive answers. A possible reason these good pairs rank the lowest influence for many validation examples could be because the queries themselves are straightforward. As seen in Table 5.7, most queries, regardless of their topic cluster, are straightforward and seek concise, specific information. They are the type of queries we might ask Siri or do a quick web search in our daily lives. This brings a second point: queries of this kind of formulation and nature may dominate the dataset. As many similar queries are inquiring about the phone number of an office or company, having hundreds of the same or almost the same training data holds almost no significance for model learning. The hard training instances, or instances that are very different from other training instances in the dataset, are more influential if we think about it.

An intriguing observation is that all these queries are derived from query-passage pairs that bear the ground truth label of relevance. Upon examining the dynamics of these pairs, we can conclude that the labels assigned in the training set are indeed accurate. Most corresponding passages either contain the exact answer or provide a comprehensive response.

One plausible explanation for these well-matched pairs consistently ranking with the lowest influence scores for many validation examples could be that the queries themselves are straightforward. As depicted in Table 5.7, a significant portion of the queries, regardless of their topic cluster, are notably direct and seek concise, specific information. These queries resemble the type of inquiries one might

PN	LD	T	D
phone number for main street sweets cedar falls	mcdermott will & emery headquarters	alex rider operation stormbreaker cast	what is functioalist perspective
salvation army muskogee ok phone number	waterloo premium outlets in waterloo	what are horseshoes made of	what is a limited company
sherwin williams phone number eden prairie	where is apple store at mayfair	buford carolina population	what is etrade platinum
phone number for elite fitness in pontotoc ms	the woodlands ice rink hours	why was the 16th amendment needed	what is nephropathy screening
heritage bank routing number hinesville ga	which province is little england located	when does mudbray evolve	what is plt in blood
phone number for dr. andrew rashkow in cody, wy			hockey what does hof mean

Table 5.7: Selected exemplary queries of the frequently encountered low influence examples by topic cluster

make to virtual assistants like Siri or conduct quick web searches in daily life.

The second point is that it is highly likely that queries formulated in this manner, characterized by their brevity and simplicity, are prevalent within the dataset. The dataset might contain many similar queries inquiring about phone numbers for various offices or companies or seeking information on locations and addresses. Consequently, training a model with hundreds of identical or nearly identical training instances reduces these similar examples' proportional influence for a prediction point. Conversely, the more influential instances are likely more difficult pairs or present stark dissimilarity to the rest of the training instances in the dataset.

5.3. Understanding Why Certain Documents are Ranked Higher for Same Query

We select a query and obtain the top 1000 retrieved documents using BM25. These 1000 documents paired for the given query correspond to the qrels in the validation dataset. Subsequently, we retrieve the scalar relevance score predictions for these qrels generated by the original model trained with the complete training set. Using these predicted relevance scores, we re-rank the 1000 documents, thereby producing a newly ordered list based on their predicted relevance scores. From this re-ranked list of documents for the chosen query, we then select the top 10 documents for visualization.

query	188714	foods and supplements to lower blood sugar
predicted relevance	ground truth label	document text
0.863	0	Low-glycemic foods that can help lower blood sugar levels include high fiber fruits, oatmeal, peanuts, beans, peas, and granola. High-glycemic foods include ... Research has shown that potatoes and white bread are converted extremely quickly into glucose.
0.849	0	Many of the foods that lower blood sugar are filling, ... will sustain your energy all day long without food cravings. The nuts ... and fish are the foods that contain omega 3 fatty acids. However, all low glycemic foods will help you to stabilize your blood sugar.

0.822	0	Foods That Safely Reduce Blood Glucose. ... there are certain foods proven to decrease blood sugar levels ... barley and black beans have been extensively studied for their ability to maintain glycemic control.
0.802	0	Cinnamon is especially rich in chromium and one of the most recommended foods for diabetics due to its ability to lower blood sugar quickly. Other great foods for your blood sugar include: beans, legumes, vegetables like broccoli and carrots...
0.792	1	Food And Supplements That Lower Blood Sugar Levels. Cinnamon: Researchers are finding that cinnamon reduces blood sugar levels naturally when taken daily. If you absolutely
0.786	0	...A 2013 review of herbal food supplements found compelling evidence that fenugreek does lower blood sugar levels in people with both type 1 and type 2 diabetes as well as those with prediabetes.
0.747	0	Low-glycemic foods that can help lower blood sugar levels include high fiber fruits, oatmeal, peanuts, beans, peas... Eating vegetables like green peas can help lower blood sugar. Avoiding drinks that are high in sugar,...
0.745	0	Cinnamon... Plant-based foods are jam-packed with fiber, which is the main reason they're so supportive of blood sugar levels. Fiber slows down the release of sugar within the bloodstream, which helps steady insulin levels.
0.711	0	Other foods for lower blood sugar levels are coconut butter, dark chocolate, cinnamon, apple cider vinegar, other nuts and seeds, most all vegetables,... and even black coffee.
0.701	0	Foods to Keep Cholesterol (and Blood Sugar) In Check. ... treating yourself to these foods can help lower your "bad" (LDL) cholesterol, boost your "good" (HDL) cholesterol ...

Table 5.8: Top 10 re-ranked documents for the selected query from validation dataset

For the query-document pairs ranked in the top 10 by the model's predictions, as illustrated in Table 5.8, we set the target label for each pair to 1 regardless of their ground truth label. This allows for an explanation as to why these documents were predicted as relevant to the query. Subsequently, we compute the TracIn influence scores for the corresponding training examples. This analysis addresses the question, "Why has the model assigned a high rank to these documents for the given query?".

Table 5.9: Some of the influential training examples observed in common for the top 10 ranked documents for the sample query. The frequency indicates how many of the 10 top documents they were evaluated within the top 10 influence. The color-coding system is as follows: blue signifies high similarity in terms of context, green indicates somewhat similar context, and gray color is used to denote unusual or problematic examples, such as those that coincidentally rank high in self-influence.)

query	document	overlap	explanation
xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	10	outlier / hard example. This example also exhibits high self-influence
immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, ...	9	related to nutrition and health

recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	8	nutrition and health. Diabetes particularly relevant to blood sugar
foods to avoid during implantation	Do not drink any caffine drinks or have chocolate... Foods containing caffeine constrict blood vessels, which can reduce blood flow to the uterus and prevent implantation of a fertilized egg...	5	nutrition and health
medication to help with severe asthma	Allergy shots (immunotherapy). Omalizumab (Xolair). This medication, given as an injection every two to four weeks, is specifically for people who have allergies and severe asthma.	4	health
what do fennel seeds taste like?	Russian Black Bread. There's a nice tangy bite to this hearty dark rye ... Cocoa and coffee powders darken the loaf, and caraway and fennel seeds impart just a bit of licorice flavor...	3	food

We examine the characteristics of the training examples that highly influence the model's ranking decision in favor of these specific documents. In our investigation, we focus on the top 10 documents ranked by the model, and for each of these, we analyze the first ten most influential training examples. The comprehensive list of these influential training examples can be found in Appendix B. A noteworthy observation is that, among these top 10 documents, there is significant overlap in the training examples that exhibit the highest influence. We report a few examples in Table 5.9. The table illustrates that we categorize our explanations into three primary groups. The first group consists of influential training examples with similar general context (highlighted in blue). The second group contains examples that exhibit context somewhat relevant to the original query context (highlighted in green). Finally, the last group does not display any discernible pattern in context similarity and is likely to be an outlier or a problematic sample (highlighted in gray). Table 5.8 illustrates that the model's top-ranked documents predominantly pertain to materials consumed to lower blood sugar levels. These documents mention foods, supplements, herbs, and their associated beneficial properties. Additionally, they often reference health-related indicators such as blood glucose levels, cholesterol, and diabetes. In blue and green groups in Table 5.9, the query document pairs from the training set exhibit a similar focus, generally around nutrition, health, or food in a broader sense.

5.3.1. Why Influence Based Subsets Were Not Better Than Random Baselines

Figure 5.9 provides valuable insights into why the approach outlined in Chapter 4 did not yield models that outperformed those trained with randomly selected training data subsets. We reconsider the ranking process summarized in Figure 5.1 to explain why influence-based subsets are not performing as expected.

The core objective of the passage ranking task is to elevate the relevant passages as high as possible within the ranked list. The ideal ranking illustrated on the right side of Figure 5.1 would position all relevant passages at the top and non-relevant ones at the bottom. For a practical example, we can

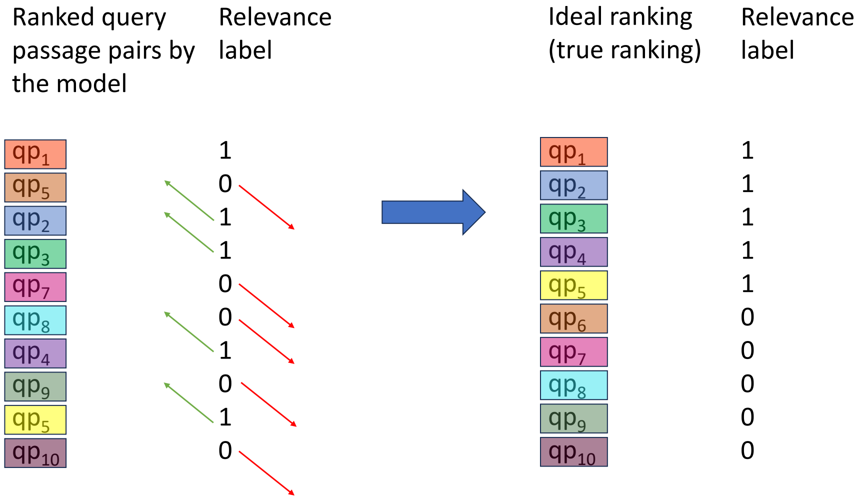


Figure 5.1: Illustration of passage ranking results. The ranked list on the left is an example model ranking passages for a given query. The ranked list on the left is the ideal ranking based on the ground truth labels. To achieve an ideal ranking, non-relevant passages in the list must move down (shown in red arrows) and relevant passages need to be pushed to the top (shown in green arrows).

consider the top 10 passages for the query ”foods and supplements to lower blood sugar,” presented in Table 5.8.

When we created subsets of the training data by retaining examples with high influence, the aim was to move a relevant passage to the top of the ranked list (depicted by the green arrows in Figure 5.1) and push non-relevant examples towards the list's bottom (the red arrows in the figure). This improvement aligns with the NDCG@10 metric, which essentially measures the positioning of relevant passages in the ranked list. A higher NDCG@10 value corresponds to a more favorable ranking, with an ideal value of 1.

However, upon examining Table 5.9, we notice that influential training data often overlaps relevant and non-relevant passages. To illustrate, if we remove the training examples highlighted in blue in Table 5.9, the rank of the relevant passage would remain unchanged since removing these influential examples also affects the order of non-relevant passages. Since proponents impact both relevant and non-relevant passages simultaneously, selecting dataset subsets by retaining proponents does not lead to meaningful improvements in ranker model performance. Further work might consider designing a mechanism that considers this overlap among proponents.

5.4. Examples with High Self-Influence

We employ another evaluation approach, self-influence, as presented in [59], to identify incorrectly labeled examples [42, 89] within the training data. This represents another potential application of instance attribution. In contrast to previous implementations, self-influence quantifies the influence of a training point on its loss. In this approach, the training point z and test point z' in the TracInCP method are identical.

According to this approach, incorrectly labeled examples tend to exert a strong influence on themselves. "Strong" in this context refers to the magnitude of the influence value, which is high. This phenomenon occurs because these examples are expected to be outliers and act as proponents since they tend to reduce the loss concerning their incorrect label [59]. Consequently, when we sort the training examples in decreasing order of self-influence, the mislabeled examples are anticipated to be ranked at the top.

Procedure:

1. We apply TracInCP to the training data using the same checkpoints as before, but this time we use the same training data as both the test and training data to measure self-influence.
2. We obtain an ordered list of the training data based on their self-influence scores. We then conduct

a human observation of the top portion of this list, analyzing the top 300 training instances, to identify mislabeled examples. The list of mislabeled examples in this list can be found in Appendix A.

3. We analyze the distribution of the self-influence values of the training instances to determine which fraction to remove from the training dataset. We subsequently retrain the model with the resulting dataset. The objective is to remove the majority of mislabeled examples.

selection	# samples	# misclassified	percentage of mislabeled examples in the list (%)
TracInCP self influence	300	71	24 %
random	300	13	4 %

Table 5.10: The amount and percentage of mislabeled training data in the selected lists of 300 highest self influence examples using TracInCP self influence approach and in comparison a randomly selected list of 300 of the data instances. The mislabeled examples are determined by human annotation. The amount of mislabeled examples in TracInCP self-influence list is significantly higher, corresponding to 24% of the list.

Training Data	Model config	val RR@10	val nDCG@10	trec19 RR@10	trec19 nDCG@10	trec20 RR@10	trec20 nDCG@10
Original training data (100%)	$\text{CrossEnc}_{\text{orig}}$	0.326	0.383	0.977	0.676	0.885	0.654
The top 10% training examples with the highest self-influence removed	$\text{CrossEnc}_{\text{orig}'}$	0.327	0.384	0.977	0.677	0.888	0.655

Table 5.11: Comparing the results between the original cross-encoder model, trained on the complete training dataset, and the Cross Encoder model trained on the dataset, with the top 10 % self-influence examples removed. The underlying hypothesis is that this process helps eliminate the majority of mislabeled examples. The evaluation metrics, RR@10 and NDCG@10, reveal only marginal improvements when mislabeled data is removed.

Table 5.10 summarizes the annotation process conducted on two sets of training data: a randomly selected 300 instances and the top 300 instances ordered by self-influence using TracInCP. The number of mislabeled examples is substantially higher in the latter group compared to the randomly selected list. This observation reinforces the argument that self-influence can serve as a valuable tool for identifying labeling errors within the training data. In Table 5.11, we compare the final evaluation metrics results of the original Cross Encoder model, trained on the complete training dataset, with the Cross Encoder model trained on the dataset after removing the top 10% of self-influence examples. The decision to remove 10 % of examples was made based on the distribution of training data self-influence values. This distribution exhibits a long tail, with the vast majority of influence scores clustering around 0 and the tail extending up to a self-influence value of 3.5. The hypothesis was that this process would help eliminate most mislabeled examples from the training data. However, the results in Table 5.11 suggest that the improvement is only marginal when the presumed mislabeled data is removed.

6 Conclusion

In this thesis, we analyzed the ability of TracInCP as an instance attribution method to identify influential examples for model predictions. We proposed using TracInCP influences for efficient dataset subset creation for passage re-ranking tasks. We applied the TracInCP influence computation method for the particular MSMARCO dataset. We used different methods to aggregate individual influence scores on a per training example-validation example prediction pairs to a meaningful influence representation of each training instance.

In this thesis, we analyzed TracInCP, an instance attribution method, to assess its efficiency in identifying influential examples for model predictions. Moreover, we proposed a novel application of TracInCP influences for efficiently generating dataset subsets tailored to passage-ranking tasks. To cater to the specifics of the MSMARCO dataset, we combined the TracInCP influence computation method with various techniques for aggregating individual influence scores across training example-validation example prediction pairs.

Influence functions have been shown to be practical tools for various applications, ranging from dataset debugging and individualized explanation generation to creating optimal data subsets. However, the effectiveness of influence values computed with instance attribution methods in IR, particularly within the passage re-ranking task, remains largely uncharted territory. Inspired by the work of Pruthi et al. [59], our research aimed to address this research gap, shedding light on the potential of instance attribution to generate significantly smaller and more efficient subsets for text-ranking tasks. Given the rise of large datasets and increasingly time-consuming training processes in AI research, this investigation becomes particularly relevant within IR and across multiple AI domains.

Our experiment uncovered several key insights. Firstly, our findings challenged the common belief - which was also our hypothesis- that selecting influential training data subsets of various sizes for ranker model training would lead to maintained performance with increased efficiency. We observed that baseline cross-encoder models trained with randomly selected data fractions demonstrated robust performance, particularly in metrics like NDCG@10 and RR@10. This suggests that pre-trained and fine-tuned cross-encoders are very robust, even when random portion data is used for fine-tuning. This observation may be data-specific and may not necessarily hold in other applications with different datasets. Our experiments on the large MSMARCO dataset suggested that random data selection results in subsets that closely mirror the distribution of the original dataset. Conversely, subsets created through influence values aggregation may not exhibit such alignment with the original distribution.

To address this distribution issue with the strong baselines, we conducted an experiment combining random data with top influence training data. Our results indicated that complete data pruning based on influence values might not be optimal for maintaining high model performance. Combining random and influence-based data subsets demonstrated improved performance over the metrics and test datasets. Still, the margin of improvement over the baseline models was not very significant. In retrieval tasks, particularly with datasets resembling MSMARCO where non-relevant query-passage pairs are not annotated in the training set, reliance on BM25 rankings for generating negative samples during model training may be problematic. Our experiments revealed that some negatives in the training set were not entirely irrelevant pairs, potentially explaining the lack of efficiency in subsets generated via instance attribution methods.

Another issue we encountered pertained to the BM25 top 1000 pairs used for negative sampling, which exhibited a high lexical overlap between queries and passages. The instance attribution method consistently assigned higher influence scores to instances with pronounced lexical overlap, even when these pairs displayed limited apparent relevance. This observation raises questions about the suitability of BM25-based negative sampling strategies for explaining retrieval models, particularly when high lexical overlap may not necessarily denote relevance.

Our qualitative analysis of influential examples unveiled the potential of TracInCP as an instance attribution method to offer valuable explanations for text ranking models. This approach enables the extraction of meaningful proponents, which, when reviewed by a human, provide logical explanations for why specific training instances supported a test instance's prediction as its true label. It is worth noting, however, that this result held for most explanations but not all. During our analysis, we encountered proponents and opponents that failed to offer meaningful insights into why a test instance received a particular prediction. This limitation could be attributed to the characteristics of the MSMARCO dataset itself, which is known for its sparsity, numerous incorrectly labeled examples, and its tendency to feature one human-labeled qrel rather than hundreds of unlabeled pairs, which may or may not all be non-relevant. Additionally, we demonstrated that instance attribution methods can address the question "What causes a certain passage to be ranked higher than another passage in the corpus?" This finding is of considerable significance, as it provides crucial insights into the collective ranking process of passages in relation to one another, moving beyond isolated explanations.

We noticed that specific queries consistently yielded low influence values, leading to their pruning from the dataset. These queries, which exhibited commonalities in terms of their topics, often contained numbers, person or facility names, and location information. The cross-encoder appears to struggle with numerical information and proper nouns in English, potentially explaining this trend.

We also introduced another valuable application of instance attribution: detecting mislabeled examples using the concept of influence. We manually annotated the top 300 examples with the highest self-influence. Past research has often suggested that examples with high self-influence are likely outliers or mislabeled instances. We showed that this was also the case for this thesis. We retrained the cross-encoder model with a modified training dataset to further validate this. This dataset was created by removing a percentage of training instances that fell in the tail end of the self-influence score distribution. The new model exhibited slightly improved NDCG@10 and RR@10 scores for both the validation and test sets.

Ultimately, the contribution of this thesis was a first and novel attempt at generating instance attribution explanations and crafting smaller, purposeful training data subsets for intricate transformer-based rankers in text retrieval scenarios. While our experiments yielded encouraging initial results, they also underscore the need for careful task-specific adjustments, such as dataset-specific considerations and methodological refinements when applying instance attribution across diverse tasks.

References

[1] Amina Adadi and Mohammed Berrada. "Peeking inside the black-box: a survey on explainable artificial intelligence (XAI)". In: *IEEE access* 6 (2018), pp. 52138–52160.

[2] Naman Agarwal, Brian Bullins, and Elad Hazan. "Second-order stochastic optimization in linear time". In: *stat* 1050 (2016), p. 15.

[3] Akshay Agrawal et al. "A rewriting system for convex optimization problems". In: *Journal of Control and Decision* 5.1 (2018), pp. 42–60.

[4] Avishek Anand et al. "Explainable Information Retrieval: A Survey". In: *arXiv preprint arXiv:2211.02405* (2022).

[5] Negar Arabzadeh et al. "Shallow pooling for sparse labels". In: *Information Retrieval Journal* 25.4 (2022), pp. 365–385.

[6] Bing Bai et al. "Supervised semantic indexing". In: *Proceedings of the 18th ACM conference on Information and knowledge management.* 2009, pp. 187–196.

[7] Krisztian Balog, Maarten De Rijke, et al. "Determining Expert Profiles (With an Application to Expert Finding)." In: *IJCAI.* Vol. 7. 625. 2007, pp. 2657–2662.

[8] Elnaz Barshan, Marc-Etienne Brunet, and Gintare Karolina Dziugaite. "Relatif: Identifying explanatory training samples via relative influence". In: *International Conference on Artificial Intelligence and Statistics.* PMLR. 2020, pp. 1899–1909.

[9] Battista Biggio, Blaine Nelson, and Pavel Laskov. "Poisoning attacks against support vector machines". In: *arXiv preprint arXiv:1206.6389* (2012).

[10] Jane Bromley et al. "Signature verification using a" siamese" time delay neural network". In: *Advances in neural information processing systems* 6 (1993).

[11] Jonathan Brophy and Daniel Lowd. "Machine unlearning for random forests". In: *International Conference on Machine Learning.* PMLR. 2021, pp. 1092–1104.

[12] Tom B. Brown et al. *Language Models are Few-Shot Learners.* 2020. arXiv: `2005.14165 [cs.CL]`.

[13] Marc-Etienne Brunet et al. "Understanding the origins of bias in word embeddings". In: *International conference on machine learning.* PMLR. 2019, pp. 803–811.

[14] Ricardo Campos et al. "Survey of temporal information retrieval and related applications". In: *ACM Computing Surveys (CSUR)* 47.2 (2014), pp. 1–41.

[15] Stefano Ceri et al. "An Introduction to Information Retrieval". In: *Web Information Retrieval.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 3–11. ISBN: 978-3-642-39314-3. DOI: `10.1007/978-3-642-39314-3_1`. URL: `https://doi.org/10.1007/978-3-642-39314-3_1`.

[16] Debaditya Chakraborty et al. "Explainable artificial intelligence reveals novel insight into tumor microenvironment conditions linked with better prognosis in patients with breast cancer". In: *Cancers* 13.14 (2021), p. 3450.

[17] Weiyu Cheng et al. "Explaining latent factor models for recommendation with influence functions". In: *arXiv preprint arXiv:1811.08120* (2018).

[18] Kyunghyun Cho et al. "On the properties of neural machine translation: Encoder-decoder approaches". In: *arXiv preprint arXiv:1409.1259* (2014).

[19] R Dennis Cook and Sanford Weisberg. "Characterizations of an empirical influence function for detecting influential cases in regression". In: *Technometrics* 22.4 (1980), pp. 495–508.

[20] Nick Craswell et al. "Overview of the TREC 2019 deep learning track". In: *arXiv preprint arXiv:2003.07820* (2020).

[21] Nick Craswell et al. *Overview of the TREC 2020 deep learning track*. 2021. arXiv: `2102.07662 [cs.IR]`.

[22] W Bruce Croft, Donald Metzler, and Trevor Strohman. *Search engines: Information retrieval in practice*. Vol. 520. Addison-Wesley Reading, 2010.

[23] Zhuyun Dai and Jamie Callan. "Context-aware term weighting for first stage passage retrieval". In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 2020, pp. 1533–1536.

[24] Arun Das and Paul Rad. "Opportunities and challenges in explainable artificial intelligence (xai): A survey". In: *arXiv preprint arXiv:2006.11371* (2020).

[25] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[26] Steven Diamond and Stephen Boyd. "CVXPY: A Python-embedded modeling language for convex optimization". In: *Journal of Machine Learning Research* 17.83 (2016), pp. 1–5.

[27] Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. "Explainable artificial intelligence: A survey". In: *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*. IEEE. 2018, pp. 0210–0215.

[28] Robert Geirhos et al. "Shortcut learning in deep neural networks". In: *Nature Machine Intelligence* 2.11 (2020), pp. 665–673.

[29] Han Guo et al. "Fastif: Scalable influence functions for efficient model interpretation and debugging". In: *arXiv preprint arXiv:2012.15781* (2020).

[30] Jiafeng Guo et al. "A Deep Relevance Matching Model for Ad-Hoc Retrieval". In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. CIKM '16. Indianapolis, Indiana, USA: Association for Computing Machinery, 2016, pp. 55–64. ISBN: 9781450340731. DOI: `10.1145/2983323.2983769`. URL: `https://doi-org.tudelft.idm.oclc.org/10.1145/2983323.2983769`.

[31] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2023. URL: `https://www.gurobi.com`.

[32] Xiaochuang Han and Yulia Tsvetkov. "Influence tuning: Demoting spurious correlations via instance attribution and instance-driven updates". In: *arXiv preprint arXiv:2110.03212* (2021).

[33] Katja Hauser et al. "Explainable artificial intelligence in skin cancer recognition: A systematic review". In: *European Journal of Cancer* 167 (2022), pp. 54–69.

[34] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[35] Samuel Humeau et al. "Poly-encoders: Transformer architectures and pre-training strategies for fast and accurate multi-sentence scoring". In: *arXiv preprint arXiv:1905.01969* (2019).

[36] Kalervo Järvelin and Jaana Kekäläinen. "Cumulated Gain-Based Evaluation of IR Techniques". In: *ACM Trans. Inf. Syst.* 20.4 (Oct. 2002), pp. 422–446. ISSN: 1046-8188. DOI: `10.1145/582415.582418`. URL: `https://doi-org.tudelft.idm.oclc.org/10.1145/582415.582418`.

[37] Zhuolin Jiang et al. "Cross-lingual information retrieval with BERT". In: *arXiv preprint arXiv:2004.13005* (2020).

[38] Jeff Johnson, Matthijs Douze, and Hervé Jégou. "Billion-Scale Similarity Search with GPUs". In: *IEEE Transactions on Big Data* 7.3 (2021), pp. 535–547. DOI: `10.1109/TBDATA.2019.2921572`.

[39] Christopher B Jones et al. "Spatial information retrieval and geographical ontologies an overview of the SPIRIT project". In: *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. 2002, pp. 387–388.

[40] Vladimir Karpukhin et al. "Dense passage retrieval for open-domain question answering". In: *arXiv preprint arXiv:2004.04906* (2020).

[41] Akhil Alfons Kodiyan. "An overview of ethical issues in using AI systems in hiring with a case study of Amazon's AI based hiring tool". In: *Researchgate Preprint* (2019), pp. 1–19.

[42] Pang Wei Koh and Percy Liang. “Understanding black-box predictions via influence functions”. In: *International conference on machine learning*. PMLR. 2017, pp. 1885–1894.

[43] Mike Lewis et al. “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension”. In: *arXiv preprint arXiv:1910.13461* (2019).

[44] Hang Li. *Learning to rank for information retrieval and natural language processing*. Springer Nature, 2022.

[45] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. “Contextualized query embeddings for conversational search”. In: *arXiv preprint arXiv:2104.08707* (2021).

[46] Tie-Yan Liu et al. “Learning to rank for information retrieval”. In: *Foundations and Trends® in Information Retrieval* 3.3 (2009), pp. 225–331.

[47] Christopher D Manning. *An introduction to information retrieval*. Cambridge university press, 2009.

[48] Tomas Mikolov et al. “Efficient estimation of word representations in vector space”. In: *arXiv preprint arXiv:1301.3781* (2013).

[49] Bhaskar Mitra, Nick Craswell, et al. “An introduction to neural information retrieval”. In: *Foundations and Trends® in Information Retrieval* 13.1 (2018), pp. 1–126.

[50] Christoph Molnar. “A guide for making black box models explainable”. In: *URL: https://christophm. github. io/interpretable-ml-book* 2.3 (2018).

[51] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[52] Tri nguyen et al. “MS MARCO: A human generated machine reading comprehension dataset”. In: *choice* 2640 (2016), p. 660.

[53] Rodrigo Nogueira and Kyunghyun Cho. “Passage Re-ranking with BERT”. In: *arXiv preprint arXiv:1901.04085* (2019).

[54] Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. “Deep learning on a data diet: Finding important examples early in training”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 20596–20607.

[55] Barak A Pearlmutter. “Fast exact multiplication by the Hessian”. In: *Neural computation* 6.1 (1994), pp. 147–160.

[56] Jeffrey Pennington, Richard Socher, and Christopher D Manning. “Glove: Global vectors for word representation”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[57] Pouya Pezeshkpour et al. “Combining feature and instance attribution to detect artifacts”. In: *arXiv preprint arXiv:2107.00323* (2021).

[58] Ronak Pradeep et al. “Squeezing water from a stone: a bag of tricks for further improving cross-encoder effectiveness for reranking”. In: *European Conference on Information Retrieval*. Springer. 2022, pp. 655–670.

[59] Garima Pruthi et al. “Estimating training data influence by tracing gradient descent”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 19920–19930.

[60] Yifan Qiao et al. “Understanding the Behaviors of BERT in Ranking”. In: *arXiv preprint arXiv:1904.07531* (2019).

[61] Yingqi Qu et al. “RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering”. In: *arXiv preprint arXiv:2010.08191* (2020).

[62] Alec Radford et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8 (2019), p. 9.

[63] Nils Reimers and Iryna Gurevych. “Sentence-bert: Sentence embeddings using siamese bert-networks”. In: *arXiv preprint arXiv:1908.10084* (2019).

[64] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “” Why should i trust you?” Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.

[65] Stephen E Robertson and K Sparck Jones. “Relevance weighting of search terms”. In: *Journal of the American Society for Information science* 27.3 (1976), pp. 129–146.

[66] Stephen E Robertson et al. “Okapi at TREC-3”. In: *Nist Special Publication Sp* 109 (1995), p. 109.

[67] Peter J Rousseeuw et al. *Robust statistics: the approach based on influence functions.* John Wiley & Sons, 2011.

[68] David E Rumelhart et al. “Sequential thought processes in PDP models”. In: *Parallel distributed processing: explorations in the microstructures of cognition* 2 (1986), pp. 3–57.

[69] Gerard Salton, Anita Wong, and Chung-Shu Yang. “A vector space model for automatic indexing”. In: *Communications of the ACM* 18.11 (1975), pp. 613–620.

[70] Xenon Market Size. “Share & COVID-19 Impact Analysis”. In: *By Application (Imaging and Lighting, Medical, satellite, Electronics&Semiconductors, and others (including R&D), and Regional Forecast, 2020–2027. Available online: https://www. fortunebusinessinsights. com/xenon-market-101965 (accessed on 9 February 2021)* (2021).

[71] Ian Soboroff, Arjen P de Vries, Nick Craswell, et al. “Overview of the TREC 2006 Enterprise Track.” In: *Trec.* Vol. 6. Citeseer. 2006, pp. 1–20.

[72] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. “Sequence to sequence learning with neural networks”. In: *Advances in neural information processing systems* 27 (2014).

[73] Simone Teufel. “An overview of evaluation methods in TREC ad hoc information retrieval and TREC question answering”. In: *Evaluation of text and speech systems* (2007), pp. 163–186.

[74] Nandan Thakur et al. “BEIR: A heterogenous benchmark for zero-shot evaluation of information retrieval models”. In: *arXiv preprint arXiv:2104.08663* (2021).

[75] Daniel Ting and Eric Brochu. “Optimal subsampling with influence functions”. In: *Advances in neural information processing systems* 31 (2018).

[76] Chau Tran et al. “Cross-lingual retrieval for iterative self-supervised training”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 2207–2219.

[77] Jack Urbanek et al. “Learning to speak and act in a fantasy text adventure game”. In: *arXiv preprint arXiv:1903.03094* (2019).

[78] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[79] Ellen M Voorhees, Donna K Harman, et al. *TREC: Experiment and evaluation in information retrieval.* Vol. 63. MIT press Cambridge, 2005.

[80] Feng Wang and David MJ Tax. “Survey on the attention based RNN model and its applications in computer vision”. In: *arXiv preprint arXiv:1601.06823* (2016).

[81] Wenhui Wang et al. “Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 5776–5788.

[82] Wenhui Wang et al. “Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers”. In: *arXiv preprint arXiv:2012.15828* (2020).

[83] Yumeng Wang, Lijun Lyu, and Avishek Anand. “BERT rankers are brittle: a study using adversarial document perturbations”. In: *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval.* 2022, pp. 115–120.

[84] Yonghui Wu et al. “Google’s neural machine translation system: Bridging the gap between human and machine translation”. In: *arXiv preprint arXiv:1609.08144* (2016).

[85] Wenhan Xiong et al. “Answering complex open-domain questions with multi-hop dense retrieval”. In: *arXiv preprint arXiv:2009.12756* (2020).

[86] Jinghan Yang, Sarthak Jain, and Byron C Wallace. “How Many and Which Training Points Would Need to be Removed to Flip this Prediction?” In: *arXiv preprint arXiv:2302.02169* (2023).

[87] Shuo Yang et al. “Dataset pruning: Reducing training data by examining generalization influence”. In: *arXiv preprint arXiv:2205.09329* (2022).

[88] Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. “Pretrained Transformers for Text Ranking: BERT and Beyond”. In: *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. WSDM ’21. Virtual Event, Israel: Association for Computing Machinery, 2021, pp. 1154–1156. ISBN: 9781450382977. DOI: 10.1145/3437963.3441667. URL: https://doi-org.tudelft.idm.oclc.org/10.1145/3437963.3441667.

[89] Chih-Kuan Yeh et al. “Representer point selection for explaining deep neural networks”. In: *Advances in neural information processing systems* 31 (2018).

[90] Shi Yu et al. “Few-Shot Conversational Dense Retrieval”. In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR ’21. Virtual Event, Canada: Association for Computing Machinery, 2021, pp. 829–838. ISBN: 9781450380379. DOI: 10.1145/3404835.3462856. URL: https://doi.org/10.1145/3404835.3462856.

[91] Ce Zhou et al. “A comprehensive survey on pretrained foundation models: A history from bert to chatgpt”. In: *arXiv preprint arXiv:2302.09419* (2023).

[92] Jianlong Zhou et al. “Effects of influence on user trust in predictive decision making”. In: *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems*. 2019, pp. 1–6.

A

High Self Influence and Mislabeled Examples

Table A.1: Selected few mislabeled data points extracted from top 300 self-influential training examples

query	doc	label	annotator label	predicted label	self inf
what class does parrot belong to	Bird Orders. Birds belong to the class Aves and live everywhere on Earth. There are about 9,000 different species of birds, divided about 30 orders...	1	0	0.223	2.466
average gas price in Nevada	There are 23 Regular gas price reports in the past 5 days in Carson City, NV. The average Regular gas price in Carson City, NV is $2.52, which is $0.27 lower than U.S. national average Regular gas price $2.79.	0	1	0.707	2.584
what is an api connector	What can API Connect do for you? IBM API Connect is a comprehensive, streamlined management solution that addresses all aspects of the API lifecycle...	1	0	0.520	2.232
list of Knoxville radio stations	List of radio stations in Tennessee. The following is a list of FCC-licensed radio stations in the U.S. state of Tennessee,...	0	1	0.925	1.931
largest mansions in the world	According to the Guinness World Records, the Imperial Palace in Beijing, China is the largest palace in the world. The Istana Nurul Iman, with 2,152,782 square feet (200,000 m2) of floorspace, holds the title as the world's largest residential palace..	1	0	0.363	1.724
price per pound for asian carp	The average price per pound of ocean fish is $6 a pound (or about $13 per kilogram) whole. This is looking at 12 to 20 cents (per pound, or about 26 to 40 cents per kilogram). So this is so affordable, he said.	1	0	0.017	1.490

what is caucasian mean	White (noun). a person with a white skin; a member of the white, or Caucasian, races of men. White (noun). a white pigment; as, Venice white. White (noun)...	0	1	0.061	1.484
what gems are loaded in rails console	Ruby is the programming language Ruby...Therefore it is good to grasp the basics of Ruby. If you just want to play with Ruby, type irb into your console to start interactive ruby...	1	0	0.021	1.434
meaning brownish colloid-like material	In the meaning of colors, brown is the color of material security and the accumulation of material possessions. The color brown relates to quality in everything-a comfortable home, the best food and drink and loyal companionship.	1	0	0.009	1.421
tours cathedral france effigies	WESTMINSTER ABBEY – LONDON. Crypts, Coronations and Royal Weddings. Westminster Abbey has been the focal point of English cultural history for a thousand years and one of the most visited tourist sites in London. Almost a million visitors a year...	1	0	0.062	1.345
how soon can you introduce baby to almond milk	Pediatricians will tell you that you should introduce nuts (tree nuts) to your baby between the age of 12 months and 36 months...	1	0	0.016	1.238
ivy university definition	The Ivy League is a collegiate athletic conference comprising sports teams from eight private institutions of higher education in the Northeastern United States. The conference name is also commonly used to refer to those eight schools as a group beyond the sports context. The eight institutions are Brown University, Columbia University, Cornell University, Dartmouth College, Harvard University, the University of Pennsylvania,...	0	1	0.721	1.187
what is a group of flies called	A collective name for a group of butterflies is called a 'Kaleidoscope'. However others have called it a 'Swarm' or 'Rabble'. In addition, the collective name for a group of caterpillars is 'an army'...	1	0	0.598	1.176

B

Example query: High influence training data for top 10 ranked documents by the model

Table B.1: Top 10 influential training examples for the top 10 ranked documents by the model. Some of the frequently occurring training examples are highlighted with color codes

doc	query	document	label
1	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	average cost of a nursing home in cincinnti ohio	Cincinnati Nursing Homes. There are 70 Nursing Homes in Cincinnati, ... The average cost of Nursing Homes in Cincinnati, OH is $201 per day.Average Cost: $201. The median cost of Nursing Homes in Cincinnati for a single-occupancy apartment is $201/day (Genworth - 2013).	1

	runescape how to bake a loaf of bread	To make bread, find a wheat field and pick some wheat. Then find a windmill, and use the wheat in the very top floor of the windmill, putting it in the hopper, and then operating the hopper controls to send it down the chute.... bucket of water on the bread will turn it into soggy bread.	1
	morning glories meaning	What's the story, Morning Glory - As sad beefor morning glory is the expresion for morning erection. ...	1
	can hotels charge for handicap parking	Valet parking is free of charge to all vehicles displaying a disabilties tag at Disney facilities offering valet service. Valet parking at the Swan and Dolphin hotels is $26 tax. ... request a validation from the hostess stand.	1
	foods to avoid during implantation	Do not drink any caffine drinks or have chocolate... Foods containing caffeine constrict blood vessels, which can reduce blood flow to the uterus and prevent implantation of a fertilized egg...	1
	medication to help with severe asthma	These include: 1 Allergy shots (immunotherapy). Over time, allergy shots gradually reduce your immune system reaction to specific allergens. 2 Omalizumab (Xolair). This medication, given as an injection every two to four weeks, is specifically for people who have allergies and severe asthma.	1
2	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	what are the best supplements for plump skin	Plumping your face to look younger can be accomplished with a few different types of fillers. Restylane, Juvederm, Perlane, Scupltra, amoung others can all fill the cheeks, temples, nasolabial folds to plump up the skin. This plumping generally makes you look much younger. As we age, we tend to lose collagen from the deeper dermis in the skin.	1
	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	runescape how to bake a loaf of bread	To make bread, find a wheat field and pick some wheat. Then find a windmill, and use the wheat in the very top floor of the windmill, putting it in the hopper, and then operating the hopper controls to send it down the chute.... bucket of water on the bread will turn it into soggy bread.	1

	what could enhance blood flow to the genital	I am looking for vitamins that I can take that will increase blood flow to my genitals... people said vitamins E, B1 (thiamine), B3 (niacin), and B12 are... show more I am looking for vitamins that I can take that will increase blood flow ...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	what do fennel seeds taste like?	Russian Black Bread. There's a nice tangy bite to this hearty dark rye ... Cocoa and coffee powders darken the loaf, and caraway and fennel seeds impart just a bit of licorice flavor...	1
	citalopram alcohol side effects	...Some antidepressants can react with alcohol. Side effects Antidepressants can have side effects such as: drowsiness, dizziness, impaired muscle co-ordination. Drinking alcohol can make these side effects worse...	1
	why do marijuana poppers make you lose weight	... Guest. i'd say it can make you lose weight, ... because when you smoke you tend to get them. I too have noticed certain cases where my friends have lost weight, it could be unrelated or related to pot, but no test suggests that pot helps you lose weight.	1
3	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	morning glories meaning	What's the story, Morning Glory - As sad beefor morning glory is the expresion for morning erection. ...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	what are the best supplements for plump skin	Plumping your face to look younger can be accomplished with a few different types of fillers. Restylane, Juvederm, Perlane, Scupltra, amoung others can all fill the cheeks, temples, nasolabial folds to plump up the skin. This plumping generally makes you look much younger. As we age, we tend to lose collagen from the deeper dermis in the skin.	1

	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	average cost of a nursing home in cincinnti ohio	Cincinnati Nursing Homes. There are 70 Nursing Homes in Cincinnati, ... The average cost of Nursing Homes in Cincinnati, OH is $201 per day.Average Cost: $201. The median cost of Nursing Homes in Cincinnati for a single-occupancy apartment is $201/day (Genworth - 2013).	1
	cost plus furniture	cost plus furniture is a furniture store that serves the little rock north little rock malvern hot springs benton areaif you are shopping for furniture ...	1
	medication to help with severe asthma	These include: 1 Allergy shots (immunotherapy). Over time, allergy shots gradually reduce your immune system reaction to specific allergens. 2 Omalizumab (Xolair). This medication, given as an injection every two to four weeks, is specifically for people who have allergies and severe asthma.	1
	diabetic management cat food	cat food feeding guide Feeding Instructions Using a standard 8-oz./250-ml measuring cup which contains approximately 144 g of Purina ® Pro Plan ® Veterinary Diets DM Dietetic Management ® Feline Formula. The following feeding program is recommended as a guideline only, with discretionary clinical adjustments for proper weight maintenance.	1
4	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1

	medication to help with severe asthma	These include: 1 Allergy shots (immunotherapy). Over time, allergy shots gradually reduce your immune system reaction to specific allergens. 2 Omalizumab (Xolair). This medication, given as an injection every two to four weeks, is specifically for people who have allergies and severe asthma.	1
	morning glories meaning	What's the story, Morning Glory - As sad beefor morning glory is the expresion for morning erection. ...	1
	what do fennel seeds taste like?	Russian Black Bread. There's a nice tangy bite to this hearty dark rye ... Cocoa and coffee powders darken the loaf, and caraway and fennel seeds impart just a bit of licorice flavor...	1
	can hotels charge for handicap parking	Valet parking is free of charge to all vehicles displaying a disabilties tag at Disney facilities offering valet service. Valet parking at the Swan and Dolphin hotels is $26 tax. ... request a validation from the hostess stand.	1
	what could enhance blood flow to the genital	I am looking for vitamins that I can take that will increase blood flow to my genitals... people said vitamins E, B1 (thiamine), B3 (niacin), and B12 are... show more I am looking for vitamins that I can take that will increase blood flow ...	1
	can gummy vitamins give you gas and bloating	Lycasin is a maltitol syrup with properties of taste and sweetness ideal... Little known to most gummy bear connoisseurs, however, the side effects of Lycasin are gas, bloating and diarrhea. In some cases the sugar-free gummy bears act as a strong laxative and leave many consumers quite uncomfortable, rushing to long trips in the bathroom.	1
5	foods to avoid during implantation	Do not drink any caffine drinks or have chocolate... Foods containing caffeine constrict blood vessels, which can reduce blood flow to the uterus and prevent implantation of a fertilized egg...	1
	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	average cost of a nursing home in cincinnti ohio	Cincinnati Nursing Homes. There are 70 Nursing Homes in Cincinnati, ... The average cost of Nursing Homes in Cincinnati, OH is $201 per day.Average Cost: $201. The median cost of Nursing Homes in Cincinnati for a single-occupancy apartment is $201/day (Genworth - 2013).	1
	what temp does al oxide melt	The melting point for aluminum foil is 660 C or 1220 F (The melting point of the foil, which is about 97% aluminum) is the same as that of aluminum. if it's made of aluminum... it melts at 660oC regardless of size or shape, but the aluminum oxide anodizing will melt at about 2000oC See the link below.	1

	cost plus furniture	cost plus furniture is a furniture store that serves the little rock north little rock malvern hot springs benton areaif you are shopping for furniture ...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	what level is wintergarden food court on	Level 2 - Currently closed for refurbishment works Level 3 - entry to food court via Bent Street-stairs to lift lobbies and O'Connell Street. Level 4 - ...	1
	how to motivate a workout	Put a mirror near your workout station! 2 Looking in the mirror while working out might give you a small boost to workout harder. 3 Try to work out with a partner....	1
	how many calories are there in flour	Calories in Almond Flour, NOW Natural Unblanched Almond Flour. Percent Daily Values are based on a 2,000 calorie diet. Your daily values may be higher or lower depending on your calorie needs.	1
	fabrication stone cost	Cost to get a new granite countertop installed varies from $30 to $75 per square foot. There are three essential factors which determine the total cost of granite countertop installation:...	1
6	foods to avoid during implantation	Do not drink any caffine drinks or have chocolate... Foods containing caffeine constrict blood vessels, which can reduce blood flow to the uterus and prevent implantation of a fertilized egg...	1
	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	why do marijuana poppers make you lose weight	... Guest. i'd say it can make you lose weight, ... because when you smoke you tend to get them. I too have noticed certain cases where my friends have lost weight, it could be unrelated or related to pot, but no test suggests that pot helps you lose weight.	1
	is hives a form of food poisoning	... Food allergies may be mistaken for food poisoning. The. most serious types of allergic reactions include sudden. itching, hives, difficulty breathing, and low blood pres-. sure. This is called anaphylaxis or allergic shock.	1

	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	geographic location of hittite empire	... publications of the British Institute of Archaeology in Ankara ; a bgn:PublicationSeries; schema:hasPart http://www.worldcat.org/oclc/404320> ;The geography of the Hittite Empire schema:name Occasional publications of the British Institute of Archaeology in Ankara ...	1
	what do fennel seeds taste like?	Russian Black Bread. There's a nice tangy bite to this hearty dark rye ... Cocoa and coffee powders darken the loaf, and caraway and fennel seeds impart just a bit of licorice flavor...	1
	what foods are good for sore muscles	... grapefruits are very good for preventing sore muscles, but yes dehydration is a big one. I injured my shoulder during a workout when I was dehydrated so make sure your drinking at least 8, 8 oz glasses of water every day.	1
	what are testosterone boosting foods	In one trial, 22 men with low testosterone levels and sperm counts were given zinc every day for 45 to 50 days. Both testosterone levels and sperm counts rose. It should not be surprising that one of the best high testosterone foods are oysters.	1
7	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	runescape how to bake a loaf of bread	To make bread, find a wheat field and pick some wheat. Then find a windmill, and use the wheat in the very top floor of the windmill, putting it in the hopper, and then operating the hopper controls to send it down the chute.... bucket of water on the bread will turn it into soggy bread.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	citalopram alcohol side effects	...Some antidepressants can react with alcohol. Side effects Antidepressants can have side effects such as: drowsiness, dizziness, impaired muscle co-ordination. Drinking alcohol can make these side effects worse...	1
	why do marijuana poppers make you lose weight	... Guest. i'd say it can make you lose weight, ... because when you smoke you tend to get them. I too have noticed certain cases where my friends have lost weight, it could be unrelated or related to pot, but no test suggests that pot helps you lose weight.	1

	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	foods to avoid during implantation	Do not drink any caffine drinks or have chocolate... Foods containing caffeine constrict blood vessels, which can reduce blood flow to the uterus and prevent implantation of a fertilized egg...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	what could enhance blood flow to the genital	I am looking for vitamins that I can take that will increase blood flow to my genitals... people said vitamins E, B1 (thiamine), B3 (niacin), and B12 are... show more I am looking for vitamins that I can take that will increase blood flow ...	1
	medication to help with severe asthma	These include: 1 Allergy shots (immunotherapy). Over time, allergy shots gradually reduce your immune system reaction to specific allergens. 2 Omalizumab (Xolair). This medication, given as an injection every two to four weeks, is specifically for people who have allergies and severe asthma.	1
8	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	why do marijuana poppers make you lose weight	... Guest. i'd say it can make you lose weight, ... because when you smoke you tend to get them. I too have noticed certain cases where my friends have lost weight, it could be unrelated or related to pot, but no test suggests that pot helps you lose weight.	1
	runescape how to bake a loaf of bread	To make bread, find a wheat field and pick some wheat. Then find a windmill, and use the wheat in the very top floor of the windmill, putting it in the hopper, and then operating the hopper controls to send it down the chute.... bucket of water on the bread will turn it into soggy bread.	1

	average cost of a nursing home in cincinnti ohio	Cincinnati Nursing Homes. There are 70 Nursing Homes in Cincinnati, ... The average cost of Nursing Homes in Cincinnati, OH is $201 per day.Average Cost: $201. The median cost of Nursing Homes in Cincinnati for a single-occupancy apartment is $201/day (Genworth - 2013).	1
	what could enhance blood flow to the genital	I am looking for vitamins that I can take that will increase blood flow to my genitals... people said vitamins E, B1 (thiamine), B3 (niacin), and B12 are... show more I am looking for vitamins that I can take that will increase blood flow ...	1
	what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
	citalopram alcohol side effects	...Some antidepressants can react with alcohol. Side effects Antidepressants can have side effects such as: drowsiness, dizziness, impaired muscle co-ordination. Drinking alcohol can make these side effects worse...	1
	how to motivate a workout	Put a mirror near your workout station! 2 Looking in the mirror while working out might give you a small boost to workout harder. 3 Try to work out with a partner....	1
9	what are the best supplements for plump skin	Plumping your face to look younger can be accomplished with a few different types of fillers. Restylane, Juvederm, Perlane, Scupltra, amoung others can all fill the cheeks, temples, nasolabial folds to plump up the skin. This plumping generally makes you look much younger. As we age, we tend to lose collagen from the deeper dermis in the skin.	1
	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	can gummy vitamins give you gas and bloating	Lycasin is a maltitol syrup with properties of taste and sweetness ideal... Little known to most gummy bear connoisseurs, however, the side effects of Lycasin are gas, bloating and diarrhea. In some cases the sugar-free gummy bears act as a strong laxative and leave many consumers quite uncomfortable, rushing to long trips in the bathroom.	1
	vitamins make me dizzy	Too much niacin may also make you feel dizzy if you get up too fast from a sitting or lying position. Large doses of niacin may make your skin flush and cause a headache, upset stomach and blurry vision, reports the University of Maryland Medical Center.	1

	foods that starve cancer cells	Certain foods, eaten in the correct portions and frequency, can provide cancer-starving benefits. Below are 5 foods to eat that can prevent cancer growth: Bok Choy This type of Chinese cabbage contains brassinin; a powerful cancer-fighter, also found in broccoli, cauliflower and Brussels sprouts.	1
	immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
	will captian morgan and diet cokes spike blood sugar	Captain Morgan Private Stock Rum 80 Proof contains 40% alcohol. Alcoholic beverages can affect your blood sugar...	1
	home remedies to lower blood sugar levels	Choose foods that are lower on the glycemic index. Eat frequent, smaller meals throughout the day every three to four hours to maintain stability for blood sugar levels. Lose weight to help manage glucose levels and lower blood sugar. Eliminate alcohol and sodas, both which cause blood sugar to fluctuate.	1
	herbs or vitamins to to increase taste buds	... it is known that zinc is required to make alkaline phosphatase, the most abundant enzyme in taste bud membranes, and zinc is also a component of a salivary protein needed for the development and maintenance of taste buds.	1
	what food to eat with iron supplement	Beef Liver. Beef is one of the most famous foods rich in iron. It is the first recommended food for people to eat when they are deficient... 100 grams of beef liver give 6.5 mg of iron, which is 36% of your daily recommended intake.	1
10	xchange leasing showroom upper marlboro md number	Sales: (240) 455-3386 | Service: (240) 455-3372. Ourisman Chevrolet Dealer Serving Upper Malboro, MD | New & Used Cars, Trucks, Vans & SUVs.	1
	recommended carbs sugar diabetics per day	Grams of Carbs. Determine the number of grams of carbs you need each day by calculating 45 to 65 percent of your total calorie intake, and dividing by 4. For example, if you eat a 2,000-calorie diet, shoot for 225 to 325 grams of carbs per day...	1
	foods to avoid during implantation	Do not drink any caffine drinks or have chocolate... Foods containing caffeine constrict blood vessels, which can reduce blood flow to the uterus and prevent implantation of a fertilized egg...	1
	morning glories meaning	What's the story, Morning Glory - As sad beefor morning glory is the expresion for morning erection. ...	1
	what foods are good for sore muscles	... grapefruits are very good for preventing sore muscles, but yes dehydration is a big one. I injured my shoulder during a workout when I was dehydrated so make sure your drinking at least 8, 8 oz glasses of water every day.	1

immune boosting foods	...It has probiotics, the good bacteria found in yogurt, some fermented foods, and your gut. Because they give your immune system a lift, they can help fight infectious diarrhea as well as other types.	1
what is a good water bottle for camp	Army Green Tactical Molle Water Bottle Hydration Pouch Bag Carrier for Hiking. Detailed introduction:MOLLE Compatible Water Bottle Pouch,With molle design ,which can be attached to any molle webbing vest, bag, duty belt or backpack. MOLLE Loops Around Entire Pouch.	1
can hotels charge for handicap parking	Valet parking is free of charge to all vehicles displaying a disabilties tag at Disney facilities offering valet service. Valet parking at the Swan and Dolphin hotels is $26 tax. ... request a validation from the hostess stand.	1
what could enhance blood flow to the genital	I am looking for vitamins that I can take that will increase blood flow to my genitals... people said vitamins E, B1 (thiamine), B3 (niacin), and B12 are... show more I am looking for vitamins that I can take that will increase blood flow ...	1
citalopram alcohol side effects	...Some antidepressants can react with alcohol. Side effects Antidepressants can have side effects such as: drowsiness, dizziness, impaired muscle coordination. Drinking alcohol can make these side effects worse...	1

C

Test metrics of the full experiment setup for varying random and influence based training dataset configurations

Figure C.1: RR@10 and NDCG@10 metrics on the trec-19 test dataset for subsets 5%, 20%, 50% and 75% constructed with different ratios of random and influence-based dataset sampling

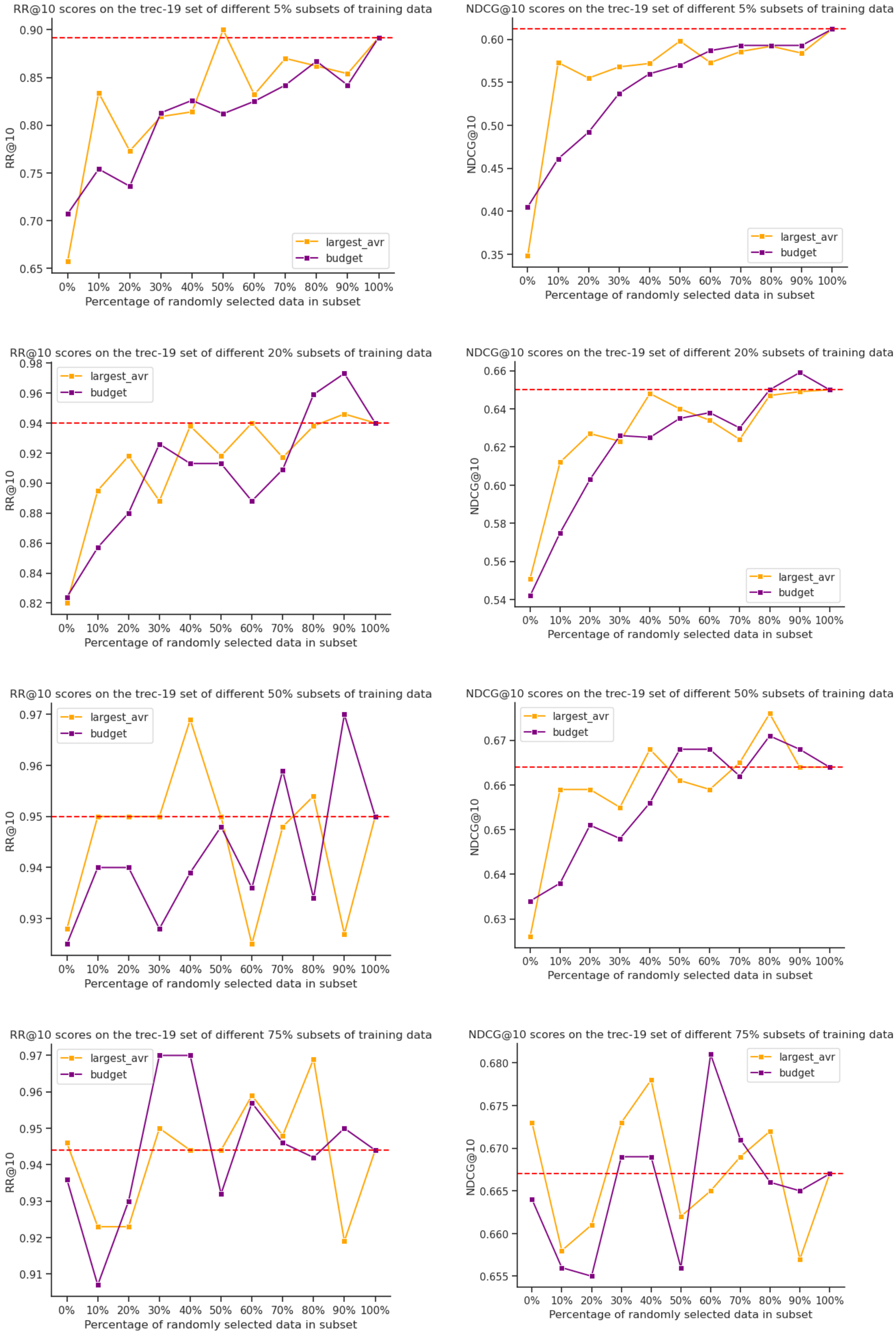
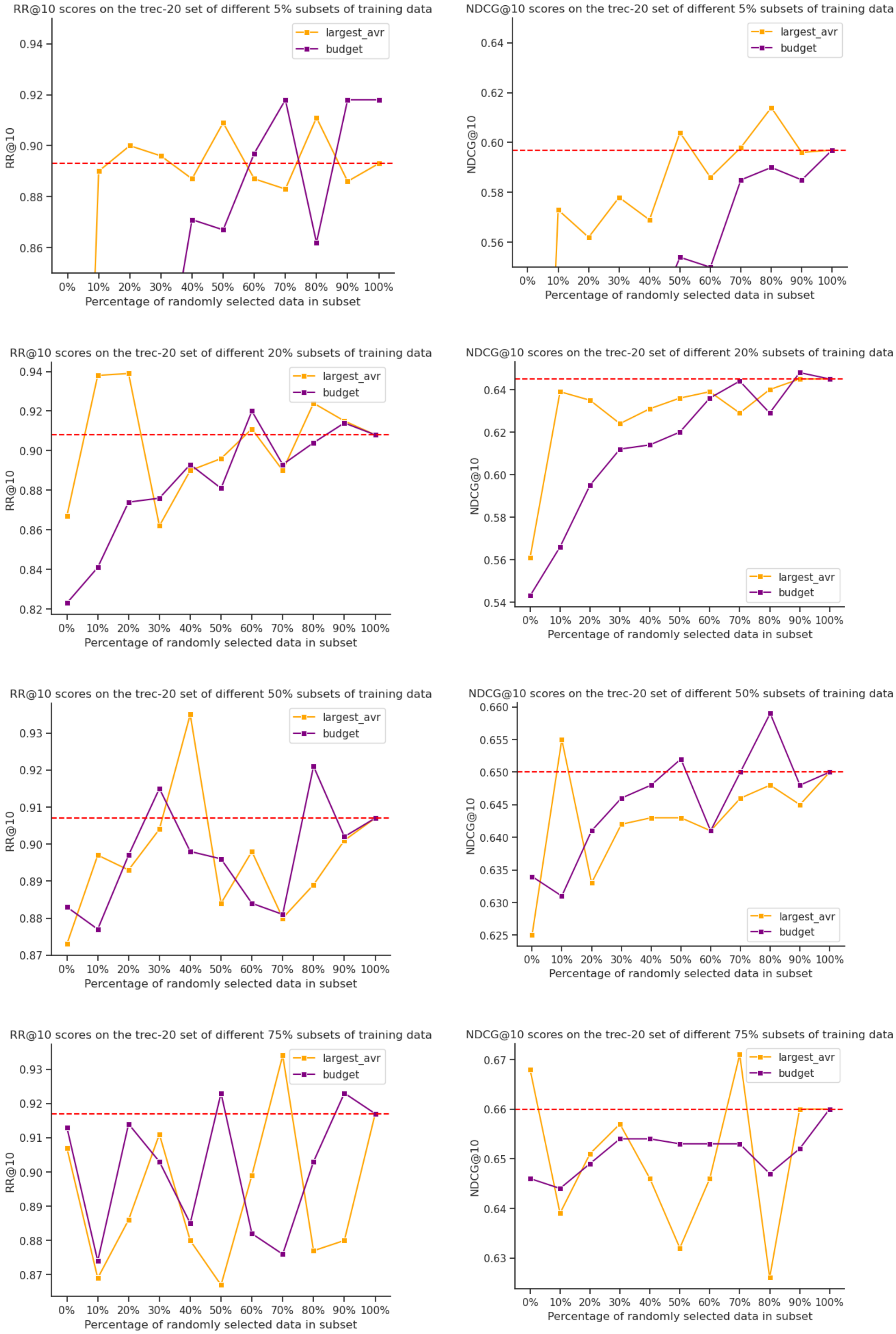


Figure C.2: RR@10 and NDCG@10 metrics on the trec-20 test dataset for subsets 5%, 20%, 50% and 75% constructed with different ratios of random and influence-based dataset sampling



RR@10 scores on the trec-20 set of different 5% subsets of training data

NDCG@10 scores on the trec-20 set of different 5% subsets of training data

RR@10 scores on the trec-20 set of different 20% subsets of training data

NDCG@10 scores on the trec-20 set of different 20% subsets of training data

RR@10 scores on the trec-20 set of different 50% subsets of training data

NDCG@10 scores on the trec-20 set of different 50% subsets of training data

RR@10 scores on the trec-20 set of different 75% subsets of training data

NDCG@10 scores on the trec-20 set of different 75% subsets of training data