NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

DIOGO LUIS EMBAIXADOR RAMOS

Master/BSc in Computer Science And Engineering

BIOMEDICAL DOCUMENT RETRIEVAL FOR DATABASE CURATION

MASTER IN COMPUTER SCIENCE AND ENGINEERING

NOVA University Lisbon
September, 2024

NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

DEPARTMENT OF
COMPUTER SCIENCE

BIOMEDICAL DOCUMENT RETRIEVAL FOR DATABASE CURATION

DIOGO LUIS EMBAIXADOR RAMOS

Master/BSc in Computer Science And Engineering

Adviser: André Lamúrias
Assistant Professor, NOVA School of Science and Technology

Examination Committee

Chair: Fernando Pedro Reino da Silva Birra
Assistant Professor, FCT-NOVA

Rapporteur: Diana Francisco de Sousa
Assistant Researcher, European Commission Joint Research Centre (JRC)

MASTER IN COMPUTER SCIENCE AND ENGINEERING

NOVA University Lisbon
September, 2024

″Le Monde ou Rien"

Acknowledgements

I want to express my sincere gratitude to my advisor, Professor André Lamúrias, for his invaluable guidance, availability, and patience throughout the development of this dissertation.

Additionally, I would like to extend my thanks to NOVA School of Science and Technology | FCT NOVA for providing me with the essential resources and support to complete this work.

Finally, to my girlfriend, family, and friends, thank you for taking the time to hear me talk about this dissertation and for the motivation they gave me.

Abstract

This dissertation explores state-of-the-art deep learning models for document retrieval in biomedical research, using the Exposome-Explorer database as a case study, which contains manually curated entries on biomarkers of exposure to environmental risk factors for various diseases. Previous works have employed simple machine learning algorithms to reduce expert workload by enhancing the accuracy and efficiency of document retrieval. In this dissertation traditional document retrieval methods, such as BM25, are evaluated alongside transformer models like MonoBERT, DistilBERT, and PubMedBERT, to assess their suitability for the task.

Results demonstrate that PubMedBERT, pre-trained on biomedical text, offers the best performance in retrieving relevant documents, with BM25 contributing significantly to initial dataset refinement. However, challenges such as curated data variability and variability in precision and recall persist, particularly with smaller datasets for which fewer training examples are available like pollutant biomarkers.

This research represents a step forward in automating and refining the curation of biomedical databases, ensuring faster and more reliable results. Future work will involve applying the trained models to the latest version of the Exposome-Explorer database and enhancing BM25 with RM3 query expansion for improved document ranking. Additional optimization of the models will be explored to address performance variability and improve overall retrieval accuracy across different biomarker datasets.

Keywords: DEEP LEARNING, DOCUMENT RETRIEVAL, DATABASE CURATION, BIOMEDICAL LITERATURE, INFORMATION RETRIEVAL

Resumo

Esta dissertação explora modelos de deep learning de última geração para a recuperação de documentos em investigação biomédica, utilizando a base de dados Exposome-Explorer como caso de estudo, a qual contém entradas manualmente curadas sobre biomarcadores de exposição a fatores de risco ambientais para várias doenças. Trabalhos anteriores utilizaram algoritmos simples de machine learning para reduzir a carga de trabalho dos especialistas, melhorando a precisão e eficiência da obtenção de documentos. Nesta dissertação, são avaliados métodos tradicionais de obtenção de documentos, como o BM25, juntamente com modelos de transformadores como MonoBERT, DistilBERT e PubMedBERT, para avaliar a sua adequação para a tarefa.

Os resultados demonstram que o PubMedBERT, pré-treinado em texto biomédico, oferece o melhor desempenho na obtenção de documentos relevantes, com o BM25 a contribuir significativamente para o refinamento inicial do conjunto de dados. No entanto, persistem desafios como a variabilidade dos dados e a variabilidade na precisão e recall, particularmente em conjuntos de dados menores, para os quais estão disponíveis menos exemplos de treino, como os biomarcadores de poluentes.

Esta investigação representa um avanço na automatização e aperfeiçoamento da curadoria de bases de dados biomédicas, garantindo resultados mais rápidos e fiáveis. Trabalhos futuros irão envolver a aplicação dos modelos treinados na versão mais recente da base de dados Exposome-Explorer e a melhoria do BM25 com expansão de consultas RM3 para um melhor ranking de documentos. Serão exploradas otimizações adicionais dos modelos para enfrentar a variabilidade de desempenho e melhorar a precisão geral da recuperação em diferentes conjuntos de dados de biomarcadores.

Palavras-chave: APRENDIZAGEM PROFUNDA, OBTENÇÃO DE DOCUMENTOS, CURADORIA DE BASE DE DADOS, LITERATURA BIOMÉDICA, OBTENÇÃO DE INFORMAÇÃO

Contents

Annexes

List of Figures

List of Tables

Acronyms

BERT Bidirectional Encoder Representations from Transformers *(pp. 1, 11–14, 22–24, 27, 32, 52, 53, 57, 58)*
BLiR Biomarker Literature Retrieval *(pp. 2, 5, 33, 35, 39, 40, 56, 62, 63, 68, 70, 71, 73–76)*
BM25 Best Matching 25 *(pp. 1, 3, 4, 6, 31, 41, 45)*
BOW Bag Of Words *(pp. 17, 19, 20)*

EHR Electronic Health Records *(p. 28)*

GB GigaBytes *(p. 19)*
GPT Generative Pre-trained Transformer *(p. 11)*
GPU Graphics Processing Units *(pp. 1, 2, 13, 74)*

IR Information Retrieval *(pp. 9, 19, 28)*

KS Keyword Summary *(pp. 30, 31)*

LM Language Model *(pp. 12, 21, 23)*
LR Logistic Regression *(p. 16)*

MLM Masked Language Model *(pp. 11, 12)*
MRR Mean Reciprocal Rank *(pp. 9, 10, 30)*
MS/Q Milliseconds per query *(p. 19)*

NDCG@10 Normalized Discounted Cumulative Gain at 10 *(pp. 9, 27, 30, 32)*
NER Named Entity Recognition *(pp. 14, 16)*
NLI Natural Language Inference *(p. 12)*
NLP Natural Language Processing *(pp. 3, 11–13, 19, 23, 52, 53, 57, 75)*
NLS Natural Language Summary *(pp. 30, 31)*

P@10 Precision at 10 *(pp. 9, 30, 32)*

RE Relation Extraction *(p. 14)*
RPrec R Precision *(pp. 9, 30, 32)*
RR@10 Reciprocal Rank at 10 *(p. 19)*

TF-IDF Term Frequency-Inverse Document Frequency *(pp. 1, 6, 16, 17, 41)*
TPU Tensor Processing Unit *(p. 13)*

WOS Web of Science *(pp. 15, 17, 39)*

1 Introduction

1.1 Context

Document retrieval involves retrieving pertinent documents from a database or document collection based on a given query. Various domains rely on databases and collections for storing information, making document retrieval crucial, especially when dealing with extensive collections comprising thousands of items. Beyond assisting users in obtaining relevant information, document retrieval plays a significant role in maintaining and curating databases by identifying entries that align with the database's relevance and purpose.

Deep learning, a subset of machine learning, uses multi-layered neural networks to learn and analyze complex data patterns. Unlike traditional machine learning, which relies on manually crafted features and performs well with smaller datasets, deep learning models transform raw data into abstract representations, automatically learning hierarchical representations of data, which allows for more nuanced and accurate pattern recognition, making them effective for unstructured data text. Techniques such as convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformer-based models like Bidirectional Encoder Representations from Transformers (BERT), J.Devlin et al [4], have revolutionized document retrieval by enhancing semantic search capabilities and context understanding.

While traditional document retrieval methods like Term Frequency-Inverse Document Frequency (TF-IDF) and Best Matching 25 (BM25), S.Robertson and H.Zaragoza [32], are still in use due to their simplicity and efficiency, deep learning approaches offer significant advantages in terms of handling complex queries and understanding context. However, they also require large datasets and significant computational power, often utilizing specialized hardware like Graphics Processing Units (GPU)s for training.

1.2 Motivation

In the present era, the advancement of deep learning algorithms, particularly versatile ones such as BERT, has led to notable successes in document retrieval and subsequent information retrieval. Deep learning models, particularly BERT-based transformers, significantly outperform simpler machine learning models by efficiently leveraging parallel computation on GPUs, enabling them to easily handle much larger datasets. These models provide more accurate results and excel at extracting complex patterns and deep semantic relationships from the data, which simpler models struggle to capture. Their ability to scale with vast amounts of data and handle intricate tasks has made them increasingly employed to enhance document retrieval across various domains, offering superior performance in data-intensive applications.

One of these domains is the biomedical domain, where maintaining relevant documents and information is critical. A case in point is the Exposome-Explorer database, dedicated to biomarkers of exposure to environmental risk factors for diseases. This database, curated manually with the analysis of around 8000 entries (documents) by experts in this domain, illustrates a scenario where deep learning models can be instrumental. These models have the potential to reduce the workload for experts by narrowing down the number of entries that they need to review while providing more precise results.

A previous study called Biomarker Literature Retrieval (BLiR), A.Lamurias et al [14], has utilized traditional machine learning models, such as Decision Tree, Logistic Regression, Naïve Bayes, Neural Network, Random Forest, and Support Vector Machine to reduce the number of documents experts needed to read. These models proved to be effective in filtering out irrelevant documents and streamlining the review process. However, more recent models, such as BERT-based models, have not yet been explored in this context as well as classical document retrieval methods such as BM25.

In biomedical document retrieval, the application of deep learning is still in its initial stages. Despite its potential, the integration of deep learning techniques has been limited due to challenges such as the scarcity of labeled training data, the complexity of biomedical data, and the need for interpretability in research and clinical settings. As a result, while deep learning has shown promise in areas like image analysis and genomic data, its implementation in biomedical document retrieval remains underdeveloped and requires further exploration and validation.

1.3 Objective

This dissertation aims to enhance document retrieval within the Exposome-Explorer database by leveraging state-of-the-art deep learning models. These advanced models promise significant improvements over the traditional machine learning techniques employed in the BLiR study. The BLiR paper demonstrated the effectiveness of using linear regression and other conventional algorithms to reduce the volume of documents

experts needed to review manually. However, deep learning models, such as transformers, offer superior capabilities in handling complex data patterns and extracting relevant information.

BERT-based models, a specific kind of deep learning models, were chosen for this research due to their superior performance in handling complex data patterns and understanding context-rich language, which is crucial for accurately retrieving and classifying relevant documents in the biomedical domain. These models have demonstrated their effectiveness in a variety of Natural Language Processing (NLP) tasks, making them ideal for the high-stakes environment of biomedical document retrieval where precision and understanding of nuanced language are paramount.

By integrating these state-of-the-art deep learning techniques, this research seeks to streamline the document examination process within the Exposome-Explorer database. The primary objective is to minimize the number of entries that require manual review by experts, thereby increasing efficiency and accuracy in identifying pertinent documents. This approach not only aims to reduce the workload for domain specialists but also to enhance the overall quality and reliability of the information retrieval process in the biomedical field. Additionally, this method can be applied to other case studies, such as databases where analyzing large volumes of documents is necessary, thereby broadening its utility.

To achieve this objective, this dissertation derived insight and inspiration from the methodologies employed in the TREC Clinical Trials competition, K.Roberts et al [31]. This competition focuses on enhancing the retrieval of clinical trial documents tailored to specific patient characteristics, aiming to optimize the patient-trial matching process. The TREC Clinical Trials competition serves as an inspiration in the field, offering a structured environment to develop, test, and refine document retrieval techniques within a real-world biomedical setting. By analyzing the strategies that have proven effective in this context, this dissertation will adapt some of these techniques to broaden the scope beyond clinical trial documents and on the objective of this dissertation.

Furthermore, the research will be informed by various studies in the biomedical domain that have addressed similar challenges. The BM25 model, a widely used probabilistic information retrieval model, was also considered for its proven effectiveness in ranking documents based on their relevance to specific queries. By integrating insights from these sources, this dissertation aims to develop a robust and efficient document retrieval system for the Exposome-Explorer database, utilizing state-of-the-art deep learning advancements.

1.4 Contributions

In terms of contribution, this dissertation presents a comprehensive model and pipeline specifically designed for the classification of relevant documents within the Exposome-Explorer database. The pipeline incorporates advanced transformer-based deep learning

models, evaluated alongside the traditional BM25 model.

To achieve this, the research leverages advanced transformer-based BERT-based models, including MonoBERT, R.Nogueira and K.Cho [22] , DistilBERT, V.Sanh et al [35] , and PubMedBERT, Y.Gu et al [11] , which have demonstrated exceptional capabilities in natural language processing tasks. These models were selected due to their ability to understand and process the intricate context found in biomedical texts, offering a substantial improvement over traditional retrieval methods.

The research methodology involves a thorough comparative analysis of the results obtained from two different approaches: BERT-based models alone and a hybrid approach combining both BM25 and BERT-based models. The hybrid approach explores the potential synergies between traditional probabilistic retrieval techniques and modern deep learning methods, seeking to capitalize on the strengths of each.

This comparative study provides a detailed evaluation of each approach's performance in terms of precision, recall, F1-score,F2-score, ROC AUC and lastly NDCG@10 . Additionally, the study examines the impact of various configurations and model adjustments, such as the use of softmax layers in MonoBERT and the tuning of hyperparameters in DistilBERT and PubMedBERT, on the overall retrieval effectiveness.

Additionally, the code developed for the models and pipeline described in this dissertation is publicly available for further research and application. The code can be accessed at the following link: Github repository. This repository includes the implementation of the transformer-based models, the hybrid BM25 approach, and all relevant scripts for data preprocessing, model training, and evaluation.

The insights gained from this analysis contribute to the field of biomedical information retrieval by clarifying the strengths and limitations of both traditional and deep learning-based approaches. The study provides a nuanced understanding of the specific contexts in which each method is most effective. Additionally, the findings offer practical considerations for the deployment of these models in real-world biomedical applications, such as improving the curation process within the Exposome-Explorer database.

1.5 Organization

This dissertation is organized into five chapters, each structured to build upon the previous content and provide a comprehensive analysis of the research conducted.

Chapter 1: Introduction

The first chapter provides an overview of the research problem, objectives, and the significance of the study. It sets the context for the work presented in the subsequent chapters, outlining the key research questions and the methodological approach employed.

Chapter 2: Related Work

The second chapter reviews the existing literature relevant to the context of this dissertation. It discusses previous work and state-of-the-art approaches, highlighting the advancements in the field and identifying the gaps that this research aims to address.

Chapter 3: The Exposome-Explorer Dataset

This chapter introduces the dataset used in this research, which consists of a collection of documents highly relevant to biomarkers of exposure to environmental risk factors for various diseases. The dataset originates from previous work within the BLiR framework, based on an earlier version of the Exposome-Explorer database, a pivotal resource for studying the connections between environmental exposures and disease risk.

Chapter 4: Document Retrieval With BM25

The fourth chapter focuses on the BM25 algorithm, which was employed as a baseline method to filter out non-relevant documents in the initial stages of the research. This chapter provides a detailed explanation of BM25, its role in this study, and its effectiveness in narrowing down the document set for further analysis.

Chapter 5: BERT Based Document Retrieval Models

The fifth chapter delves into the exploration and application of various BERT-based models, including MonoBERT, DistilBERT, and PubMedBERT. It examines the performance of these models within the context of this research, with particular attention to their ability to enhance document retrieval and classification. The chapter also addresses the specific challenges encountered when using these models, such as computational limitations and the adjustments made to improve their performance.

Chapter 5: Final Remarks

The final chapter summarizes the findings of the research, discusses the implications of the results, and outlines potential directions for future work. It reflects on the overall contribution of the dissertation to the field and provides a closing assessment of the research objectives.

2

Related Work

2.1 Document Retrieval

Understanding the motivation for this dissertation requires acknowledging the era before transformers and contemporary machine learning. In this period, document retrieval, the process of obtaining relevant documents based on user queries, heavily relied on traditional methods. One pivotal approach was BM25, considered foundational for such tasks.

2.1.1 BM25-Based Document Retrieval

Best Matching 25 (BM25) , S.Robertson and H.Zaragoza [32], is a ranking function commonly used in information retrieval and search engines to estimate the relevance of a document to a given search query. It is an extension of the BM (Best Matching) model, designed to overcome some limitations of traditional Term Frequency-Inverse Document Frequency (TF-IDF) weighting. It extends term frequency by using strategies to avoid giving too much importance to high-frequency values; while observing that if a term appears more frequently in a document, its importance in determining relevance decreases which differentiates itself from other models. Another very important feature is Document Length Normalization which makes longer documents not have an advantage over smaller documents by making them penalized for being bigger.

The BM25 score of a document d for a given query q is given by:

$$BM25 = \sum_{t \in q} log[\frac{N}{df(t)}] \cdot \frac{(k_1 + 1) \cdot tf(t,d)}{k_1 \cdot [(1-b) + b \cdot \frac{dl(d)}{dl_{xxx}}] + tf(t,d)}$$

In the equation, the symbol t represents a term within the query q, iterating over all terms in the query. N represents the total number of documents available in the document collection or corpus. $df(t)$ signifies the document frequency of term t, indicating how many documents in the collection contain term t. $tf(t,d)$ denotes the term frequency of term t in document d, representing how many times term t appears in document d. dl stands for

the length of document d, representing the total number of terms in the document. Lastly, dl_{xxx} denotes the average length of documents across the corpus. BM25 offers flexible customization through two key parameters: k1 and b. These parameters enable users to fine-tune the model to suit the specific properties of their data best. k1 governs the saturation level, with higher values amplifying the saturation effect, meaning it influences how quickly the normalization of term frequencies saturates as term frequencies increase. b, on the other hand, dictates the extent of document length normalization. When b approaches 1, normalization is less pronounced, while values closer to 0 introduce a stronger normalization effect.

BM25 provides flexibility through its adjustable parameters, k1 and b, empowering users to customize the model according to the specific characteristics of their data. This adaptability and its reliable performance have established BM25 as a fundamental component of state-of-the-art information retrieval systems. Despite considering critical factors like document size, these traditional methods were deemed ineffective in the general domain. Recognizing these limitations prompted the exploration and introduction of innovative methods to enhance information retrieval performance.

2.1.2 Metrics

In the context of measuring results, several key metrics are employed to quantify the performance of machine learning models, particularly in document retrieval tasks. These include Precision, Recall, F1 Score, F2 Score, and ROC AUC. Each metric serves a unique purpose and provides insight into different aspects of the model's predictive capabilities.

Precision is a metric that quantifies the accuracy of positive predictions made by the model. It is defined as the proportion of true positive results (correctly identified relevant documents) to the total number of documents predicted as positive, which includes both true positives (**TP**) and false positives (**FP**). Precision is essential in cases where the primary concern is minimizing false positives. In document retrieval systems, high precision indicates that when the model identifies a document as relevant, it is very likely to be truly relevant. Mathematically, Precision is expressed as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{2.1}$$

Recall, also known as sensitivity or true positive rate, measures the model's ability to correctly identify all relevant instances within a dataset. It is defined as the ratio of true positive predictions to the sum of true positives (**TP**) and false negatives (**FN**). Recall is particularly important in scenarios where missing relevant documents (false negatives) is more costly than incorrectly identifying irrelevant ones (false positives). A high recall value indicates that the model is effective in identifying most of the relevant documents, even if some irrelevant ones are also captured. The formula for Recall is given by:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.2}$$

The **F1 Score** is the harmonic mean of Precision and Recall, offering a balanced measure that considers both false positives and false negatives. It is particularly useful when the cost of false positives and false negatives is equally important. The F1 score ranges from 0 to 1, where a value close to 1 indicates a strong balance between Precision and Recall. This score is critical when one seeks to balance the trade-off between the two metrics in document retrieval, ensuring both accuracy and completeness. The F1 Score is mathematically defined as:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.3}$$

The **F2 Score** is a variation of the F1 Score that places more emphasis on Recall than on Precision. It is beneficial in situations where identifying all relevant documents is more critical than ensuring that the predicted relevant documents are indeed correct. This is particularly useful in biomedical applications where missing a relevant document could lead to significant consequences.This metric amplifies the importance of Recall, thus prioritizing the identification of as many relevant documents as possible, even if it comes at the expense of precision. The F2 Score is given by:

$$F2 = (1 + 2^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(2^2 \cdot \text{Precision}) + \text{Recall}} \tag{2.4}$$

The **Receiver Operating Characteristic** (ROC) curve is a graphical plot that illustrates the performance of a classification model across different threshold settings. The area under the ROC curve (AUC) provides a single value that summarizes the model's ability to distinguish between positive and negative classes. The ROC curve plots the true positive rate (Recall) against the false positive rate (FPR), defined as:

$$\text{FPR} = \frac{FP}{FP + TN} \tag{2.5}$$

The **AUC** is calculated as the integral of the ROC curve, providing a measure of how well the model can separate the two classes. A model with a higher AUC value has a stronger ability to differentiate between relevant and irrelevant documents. The AUC can be expressed as:

$$\text{AUC} = \int_0^1 TPR(FPR)\, dFPR \tag{2.6}$$

ROC AUC is a valuable metric in document retrieval as it reflects the model's overall discriminatory power, independent of any particular decision threshold.

By employing these metrics a comprehensive insight into the performance of document retrieval models can be gained. Each metric provides a different perspective, allowing one to balance the trade-offs between identifying relevant documents and excluding irrelevant ones. These metrics offer a robust framework to assess and refine the model's accuracy, particularly in biomedical document retrieval.

2.1.2.1 Metrics used in TREC competition

The metrics used to measure TREC competition results are Normalized Discounted Cumulative Gain at 10 (NDCG@10), Precision at 10 (P@10), R Precision (RPrec), and Mean Reciprocal Rank (MRR). These metrics are widely used in Information Retrieval (IR) tasks, with the NDCG@10 being used to compare and evaluate the work performed in this dissertation and the TREC competition.

NDCG@10 provides a measure of the quality of a ranked list of documents, giving more weight to highly relevant documents at the top of the list while accounting for the diminishing returns of relevance as it progresses down the list.

$$\text{NDCG@10} = \frac{\text{DCG@10}}{\text{IDCG@10}} \tag{2.7}$$

where:

- DCG@10 is the **Discounted Cumulative Gain at 10**, calculated as:

$$\text{DCG@10} = \sum_{i=1}^{10} \frac{2^{\text{rel}_i} - 1}{\log_2(i+1)} \tag{2.8}$$

  Here, rel_i is the relevance score of the document at position i.

- IDCG@10 is the **Ideal Discounted Cumulative Gain at 10**, which is calculated in the same way as DCG@10 but with the documents sorted by relevance in descending order, ensuring the highest possible score.

P@10 is a metric used to evaluate the relevance of documents returned by a search or information retrieval system. Precision measures the proportion of relevant documents among the top k documents in the ranked list. In the case of P@10, it specifically looks at the precision at the top 10 positions of the ranked list.

$$\text{P@10} = \frac{\text{Number of relevant items in the top 10}}{10} \tag{2.9}$$

RPrec is a metric used to evaluate the performance of an IR system, particularly in the context of ranked retrieval. It measures the precision of a search algorithm by considering the top R documents, where R is the total number of relevant documents for a given query.

$$\text{RPrec@k} = \frac{\text{Number of relevant documents at ranks } \leq k}{k} \tag{2.10}$$

MRR is a single-valued metric that provides an overall measure of how well the system, on average, places the first relevant item in the ranked list. A higher MRR indicates better performance, as it implies that relevant items tend to appear at higher ranks in the list.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \tag{2.11}$$

In this equation, Q represents the set of queries, and $|Q|$ is the total number of queries. The term rank_i refers to the rank position of the first relevant item for the i-th query. Therefore, the MRR computes the average of the reciprocal ranks of the first relevant result across all queries. A higher MRR indicates that relevant items generally appear at higher positions in the ranked results.

2.2 Deep Learning in NLP

2.2.1 Transformer models

When tackling the complexities of natural language tasks a transformer architecture has been adopted, A.Vaswani et al [38]. Unlike its predecessors, this type of model exclusively relies on attention mechanisms, eliminating the need for recurrent and convolutional techniques. This strategic decision not only improves parallel processing but also significantly reduces overall training time, making the model more efficient and scalable.

The transformer architecture stands out due to its unique ability to handle global dependencies in a parallelized manner, overcoming the sequential limitations inherent to recurrent models. These recurrent models introduced a sequential nature, forcing dependencies to adhere to a predefined order. Despite incremental improvements, their inherent sequential processing hampers effective parallelization, limiting the model's ability to capture global dependencies simultaneously. While convolutional algorithms permitted both sequential and parallel processing of dependencies, they encountered difficulties in relating signals between distant input and output positions.

A pivotal element of the transformer model is the incorporation of attention mechanisms, which enhance efficiency in capturing global dependencies. An attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key. This compatibility function is a scalar dot-product, where q is a query vector and k is the key vector:

$$\text{Attention Score}(q, k_i) = q \cdot k_i \tag{2.12}$$

The attention mechanism used in the model is referred to as Scaled Dot-Product Attention. This mechanism operates on input consisting of queries and keys, each with a dimension of dk(k = keys), and values with a dimension of dv(v = values). The process involves computing dot products between the queries and all keys, dividing each by the square root of dk, and applying a softmax function to determine weights for the corresponding values.

In practical terms, the attention function is computed simultaneously for a set of queries, which are organized into a matrix Q. Keys and values are also organized into matrices K and V. The output matrix is then computed as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.13}$$

The practical application of attention mechanisms within the proposed model involves several crucial steps. These include the transformation of input sequences through sentence splitting and tokenization, numerical representation through word embedding, and the use of multi-head attention mechanisms for generating key-value pairs. These steps collectively contribute to the model's ability to efficiently capture and leverage global dependencies in the input. By prioritizing attention mechanisms and their efficient parallelizability, this model presents a state-of-the-art solution to challenges posed by traditional neural network architectures.

2.2.2 BERT

Bidirectional Encoder Representations from Transformers (BERT), J.Devlin et al [4], is a deep learning model from Google that pretrains comprehensive bidirectional representations from unannotated text. This is achieved by simultaneously considering both left and right context across all layers of the model. Consequently, the pre-trained BERT model can undergo fine-tuning by adding a single output layer, enabling the creation of cutting-edge models for various tasks, including question answering and language inference, without requiring significant task-specific architectural adjustments. Due to the effectiveness of pre-trained language representations, two strategies exist for utilizing them which are the feature-based and the fine-tuning approaches. Feature-based methods, like ELMo, M.E.Peters et al [24], integrate pre-trained representations as additional features within task-specific architectures. Fine-tuning, exemplified by Generative Pre-trained Transformer (GPT), A.Radford et al [29], minimally introduces task-specific parameters and fine-tunes all pre-trained parameters on downstream tasks. However, unidirectional language models in both approaches limit representation power, particularly for tasks requiring bidirectional context understanding, posing challenges for token-level tasks like question answering. BERT is introduced to enhance fine-tuning approaches. BERT overcomes unidirectionality by using a Masked Language Model (MLM) objective, allowing bidirectional pre-training. Unlike previous models,BERT reduces the need for complex task-specific architectures and achieves state-of-the-art performance across various Natural Language Processing (NLP) tasks, outperforming prior approaches.

BERT's implementation consists of two steps which are pre-training and fine-tuning. During pre-training, the model learns from unlabeled data across various tasks. In fine-tuning, BERT uses its pre-trained parameters and adapts them to specific tasks using labeled data. Each task has its own fine-tuned model, initiated with the same pre-trained parameters.

2.2.2.1 Input-Output Representations

BERT is designed to effectively manage diverse downstream tasks by enabling its input representation to represent either a single sentence or a pair of sentences within a single token sequence. In this context, a sentence does not necessarily refer to a grammatical sentence in natural language; rather, it refers to a sequence of tokens that are fed into the BERT model for processing. BERT uses WordPiece, Y.Wu et al [41], tokenization where token input representations are constructed by summing token, position, and segment embeddings.

2.2.2.2 Pre-training BERT

BERT was pre-trained using two unsupervised tasks, MLM and Next Sentence Prediction.

MLM is a technique devised to address the challenges encountered when training bidirectionally. Its bidirectional nature poses a problem as tokens can indirectly reveal information from the other side, rendering masking impossible. MLM removes these problems by masking a certain percentage of input tokens and then predicting what these masked tokens are by taking into account the context and words around them.

Next Sentence Prediction is a technique designed to help the model understand the connection between two sentences, consisting of a binary next sentence prediction task, easily generated from any monolingual corpus.

BERT was pre-trained using data from BooksCorpus (800M words) and English Wikipedia (2,500M words) to apply these techniques and improve its efficiency in NLP tasks.

	MNLI-m (Accuracy)	QNLI (Accuracy)	MRPC (F1)	SST-2 (Accuracy)	SQuAD (F1)
BERTBASE	84.4	88.4	86.7	92.7	88.5
No NSP	83.9	84.9	86.5	92.6	87.9
LTR & No NSP	82.1	84.3	77.5	92.1	77.8

Table 2.1: Performance of BERT using different techniques, J.Devlin et al [4]

All of the columns in the table above (2.1) are different datasets such as SQuAD which is the Stanford Question Answering Dataset and SST-2 is the Stanford Sentiment Treebank. MNLI-m corresponds to a Multi-genre Natural Language Inference (NLI) matched dataset aiming to predict whether a given premise sentence entails, contradicts, or is neutral concerning a given hypothesis sentence. QNLI corresponds to Question-answering NLI and MRPC is the Microsoft Research Paraphrase Corpus.

The influence of pre-training steps is depicted above (see Table 2.1), where "No NSP" indicates the absence of Next Sentence Prediction, negatively affecting performance. Additionally, "LTR" signifies it operates as a left-to-right Language Model (LM), which has a more pronounced impact.

2.2.2.3 Fine-tuning BERT

For any given task, the fine-tuning process involves inserting task-specific layers into BERT and training all parameters end-to-end. In the input, sentence A and sentence B from pre-training correspond to various scenarios, such as sentence pairs in paraphrasing, hypothesis-premise pairs in entailment, question-passage pairs in question answering, and a degenerate text-pair in text classification or sequence tagging. In the output, token representations are directed to an output layer for token-level tasks like sequence tagging or question answering, while the [CLS] representation is directed to an output layer for classification tasks like entailment or sentiment analysis.

The process is cost-effective, with the cost of replicating the original results taking less than an hour on a single Cloud Tensor Processing Unit (TPU) or a few hours on a Graphics Processing Units (GPU) from the same pre-trained model.

2.2.2.4 BERT Performance

System	Dev F1	Test F1
Fine-tuning approach		
BERT_LARGE	96.6	92.8
BERT_BASE	96.4	92.4
Feature-based approach (BERT_BASE)		
Embeddings	91.0	-
Second-to-Last Hidden	95.6	-
Last Hidden	94.9	-
Weighted Sum Last Four Hidden	95.9	-
Concat Last Four Hidden	96.1	-
Weighted Sum All 12 Layers	95.5	-

Table 2.2: System and F1 Scores, J.Devlin et al [4]

In the provided table (2.2), the comparison between the fine-tuning and feature-based approaches underscores why BERT was developed specifically for fine-tuning.

BERT initially achieved state-of-the-art results in 2019, setting the standard for NLP, and serving as the foundation for various models, including BioBERT (see Section 2.2.3). However, for the scope of this dissertation, BERT serves as a starting point, excelling in general domain tasks rather than the biomedical domain required. Hence, more specialized models like BioBERT were created to enhance performance in biomedical-specific tasks.

2.2.3 BioBERT

This paper, J.Lee et al [17], introduces a BERT-based model called BioBERT, which was done to increase the performance of BERT, J.Devlin et al [4], in the biomedical domain. The base model selected was BERT due to its strong performance in NLP tasks. However,

since BERT was trained on general internet data, it is more suited for everyday language rather than the specialized biomedical language used in this study.

This model was trained using either abstracts or full-text articles from PubMed to prepare this model to better accommodate biomedical words and context. Only a few changes were made to the structure of BERT, as, the tokenizer used by the BioBERT, WordPiece is the same as BERT since new words are represented by their sub-words, which eases their integration.

BioBERT distinguishes itself from other models in biomedical Named Entity Recognition (NER) by fine-tuning word embeddings specifically for biomedical tasks after being pre-trained on large biomedical corpora, such as PubMed, instead of relying solely on general-purpose embeddings. Another difference is that to improve Relation Extraction (RE) performance, it utilizes target entities specific to biomedical terms such as Gene and Disease.

Another noteworthy aspect to take into account is that the selection of datasets that are used are relevant to the domain-specific task and also used in other similar papers, S.Lim and J.Kang [18], G.Tsatsaronis et al [36], A.Vaswani et al [39]. This deliberate choice serves a dual purpose: firstly, it ensures the relevance of the datasets to the specific biomedical domain addressed by Bio-BERT, and secondly, it facilitates a comparative analysis with results from other similar papers. By using shared datasets (NBCI disease dataset , R.Dogan and R.Leaman and Z.lu [5] and i2b2/VA dataset, Ö.Uzuner et al. [37]), BioBERT establishes a common ground for evaluation, allowing for a comparison of its performance against that of other models in the field.

The study underscores the importance of increased text input for boosting model performance and highlights how varying pre-training steps impact performance variability. These conclusions are crucial for my dissertation, offering a foundational understanding and suggesting optimal parameter values for potential model improvements. The diverse model configurations provide a framework for experimentation in my research, allowing for systematic parameter adjustments and exploration of different architectures to develop a better-performing model. BioBERT's tailored approach to the biomedical spectrum serves as a guide for training models in domain-specific tasks, offering insights into expectations and potential adjustments during research, though staying updated on newer papers is advised for the latest advancements.

2.3 Exposome-Explorer Use Case

Exposome-Explorer [6] is the first database specialized in biomarkers of exposure to environmental risk factors for diseases. Manually curated by specialists, the database provides detailed information on biomarkers, including their nature, measured populations, biospecimens, methodologies, concentrations, correlations, and reproducibility.

Researchers can compare biomarker performance and identify valuable ones for biomonitoring and disease studies. Leveraging these insights enables informed decisions, enhancing precision and effectiveness in biomarker selection and utilization for investigations in biomonitoring and disease causation studies. The depth of information highlights the database's relevance in advancing research in these critical areas. With each new update to the database, more entries are added with the current number being about 1212 entries in the biomarkers section. Being manually curated, specialists have spent a lot of time reviewing more than 8500 citations, in the first release, considering that in the latest release about 1056 citations were considered relevant. As such, to aid experts and to avoid such extensive reading by them, this dissertation proposes to use deep learning models to save time and reduce the number of articles analyzed by the experts.

2.4 BLiR

This section is centered on the paper, A.Lamurias et al [14], identified as a prior work sharing the objective of this dissertation, as discussed in Section 2.3 . Both the paper and this dissertation aim to improve the process of literature review by minimizing the volume of papers scrutinized by experts. The distinction lies in the methodology employed; while the referenced paper utilizes machine learning models, this dissertation will leverage deep learning algorithms. Experts will still need to read articles and curate them manually but the amount of citations is expected to be highly reduced. A lot of work can be taken from this paper which can help and give insights in how to deal with the data obtained.

This paper's results were that out of the 905 dietary publications employed for classifier testing, 365 were categorized as positive. This reduction of 66% in the proportion of articles classified as positive would lead to a time-saving of 77.9% (recall score) for identifying relevant articles, with only 22.11% of the pertinent articles being potentially overlooked.

2.4.1 Initial Data

To replicate the methodology employed by experts that curate the Exposome-Explorer, the queries that they initially used were executed on the Web of Science (WOS) [40] to identify biomarkers. These queries, important to the subsequent stages, were not only employed for the search but were also stored as supplementary data, serving as inputs for certain models to enhance their performance.

The search yielded approximately 8575 citations, which were then subjected to classification as either relevant or non-relevant. To validate the results, a subset of 480 citations, deemed relevant by the Exposome-Explorer team, was utilized as a benchmark.

However, it is noteworthy that approximately 84 publications, identified through additional efforts by the Exposome-Explorer experts, were absent from the WOS search results. This means that these publications could not be included in the dataset due to the inherent challenge of replicating the original Exposome-Explorer process.

Further work was done to account for the lack of certain fields such as PubMedID. To address this gap, PubMed served as a resource for extracting titles, abstracts, and metadata, encompassing publication dates, author names, citation frequency, and journal details. The retrieval of PMIDs, titles, abstracts, and metadata from PubMed was facilitated using E-utilities, a public API accessible through the NCBI Entrez system. To compile a comprehensive dataset, some publications were identified using the DOI to PMID (PubMed ID) converter, while others were located through a combined search utilizing the article title and the first author's name. This final approach resulted in a corpus of 7,083 publications.

Text preprocessing was performed to adapt the data to serve as input to the machine learning algorithms, using techniques such as tokenization, stematization, and stop word removal. Additional steps included assigning labels to each article, building matrixes that represent either n-gram counts or TF-IDF count, and transforming text to numerical data.

2.4.2 Models

Several machine learning models were built, however, in the context of this dissertation, only the neural network built will be taken into account. The neural network was a very limited one having only around 100 neurons, showing once again how much can be improved by the use of more powerful models. Ensemble methods were also used specifically bagging(training a classifier on randomly selected subsets of the training data, and subsequently, the outcomes are aggregated) and stacking (training of multiple classifiers, and their outputs are utilized to train a final model that makes predictions for the respective classes).

A NER model was also made, to provide experts with a way to search for biomarkers present in articles. As a dataset for this kind of task was not provided, data were obtained by searching for biomarker entities in the documents, using MER, F.Couto and A.Lamurias [2].

2.4.3 Results

After obtaining the results through testing it was determined that the most efficient combination was by using titles plus abstracts. By examining the outcomes derived from the titles and metadata dataset, lower overall values were observed in comparison to the abstract dataset. Incorporating features from both titles and abstracts yielded improved F2 scores across nearly every algorithm, surpassing the performance achieved when using these features individually. This suggests that akin to the manual curation process, assessing the relevance of an article to the database should consider both titles and abstracts.

The results (Presented in table 2.3) presented above point to the Logistic Regression (LR) algorithm displaying superior performance across numerous metrics. The Neural Network algorithm, on the other hand, attained the highest precision when employing abstracts,

	Precision	Recall	F1	F2	AUC
DT	0.403	0.532	0.459	0.500	0.744
LR	0.530	0.745	**0.619**	**0.689**	0.864
NB	0.515	0.745	0.609	0.684	0.853
NN	**0.700**	0.447	0.545	0.482	0.718
RF	0.450	0.766	0.567	0.672	0.657
SVM	0.451	0.787	0.574	0.685	0.867
Bagging	0.542	0.681	0.604	0.648	0.825
Stacking	0.388	**0.851**	0.533	0.687	**0.889**

Table 2.3: BLiR Models Results, A.Lamurias et al [14]

titles, and abstracts, and titles and metadata datasets. This shows that deep learning models have the potential to improve and need to be explored deeper, demonstrating the purpose of this dissertation.

This paper, being previous work on the subject, is considered important to the work that will be done in this dissertation, allowing for a previous insight into the objective. Relevant sections will be put to use in this dissertation, like using the dataset that includes the WOS result and how to simulate the process of manually curating the data by the Exposome-Explorer experts.

2.5 Document Retrieval Models

2.5.1 Sparse and Dense Retrieval Methods

To better understand these different document retrieval methods the section below will focus on the differences between sparse and dense document retrieval methods.

Sparse, as the name says, relies on the use of sparse vectors which means that most of the entries are zero. This is the case of the Bag Of Words (BOW) approach, which counts the frequency of words and constructs a vector with it, with many values being possibly zero from its absence. Other relevant things about sparse are their use of cosine similarity to determine the similarity between documents, TF-IDF that weights the importance of certain terms in each document, as well as the scalability, issues it presents due to significant computational costs. Dense methods use continuous vector representations, obtained through either word or document embeddings to capture context and relationships between documents and words. Contrary to sparse, they are low-dimensional, as they do not have zero values available meaning representations can be smaller. Another big difference is the leverage of transformers, A.Vaswani et al [38] to learn the context and relationships between words, making the scalability of these methods potentially more complex in terms of computational resources. In summary, sparse document retrieval methods rely on high-dimensional sparse representations

which are based on word frequency, while dense document retrieval methods use low-dimensional, dense vector representations obtained through embeddings, often providing better semantic understanding but at a potentially higher computational cost.

2.5.1.1 DocT5Query

Doc2Query is a neural model designed for generating queries from a given document, utilizing a sequence-to-sequence architecture. This architecture consists of using sequences of tokens in both the input and output, to map an input sequence into an output sequence. It learns to generate queries that capture the information present within the document, improving information retrieval and question-answering tasks. DocT5query, designed as a sparse retrieval method for the initial document retrieval phase, produces a set of queries related to a particular text, attaching them to the document, which can be matched with the input query to select which documents to retrieve. Nevertheless, an issue arises due to its sequence-to-sequence nature, as it often produces queries that are insufficiently aligned with the provided text, making them not eligible to be used.

The relevance scoring function is utilized to filter queries based on their relevance score, ensuring that only those exceeding a predetermined threshold t are retained, with D′ being the new corpus that was produced by the concatenation of the corpus D with the expansion queries and e being expansion function that maps a document to queries:

$$D' = \text{Concat}(d, (q|q \in e(d) \wedge s(q,d) \geq t))|d \in D)$$

DocT5query–, M.Gospodinov and S.MacAvaney and C.Macdonald [10], differentiates itself from the standard Doc2query by using a T5, C.Raffel et al [30], deep learning model and by having an extra phase compared to DocT5query, which is the filtering phase. The generation phase, as the name indicates produces a collection of n queries that might be responded to by the document. The filtering phase consists of filtering the inadequate queries generated, removing them, and appending the ones that are left in the document.

$$D' = \{\text{Concat}(d, e(d)) \mid d \in D\}$$

is the formula corresponding to the output of Doc2query, with the rest being the new addition provided by DocT5Query, like the threshold t which depends on the relevance scoring function. This study employs a hybrid approach, utilizing the distribution of relevance scores across all expansion queries to determine the optimal value of t, meaning that with a relevance score threshold of 0.3, only queries falling within the top 30% are retained.

To perform the experiments, Doc2query using the T5 transformer was used, D.R.Cheriton. [1], as to the date of the paper it was the best model, as well as three different methods to perform the query filtration: ELECTRA5, R.Pradeep et al [27], MonoT5 , R.Pradeep, R.Nogueira, and J.J.Lin [25], and TCT-ColBERT7 , S.-C.Lin, J.-H.Yang, and J.Lin [19], which

are in the same order, with the two first being cross-encoder models and the last one a dual-encoder model.

System		RR@10		nDCG@10		ms/q	GB
Dataset	Dev	Dev2	Eval	DL'19	DL'20	MRT	Index
BM25	0.185	0.182	**0.186**	0.499	0.479	5	0.71
Doc2Query (n = 40)	0.277	0.265	0.272	0.626	0.607	30	1.17
w/ ELECTRA Filter (30%)	**0.316**	**0.310**	**0.292**	**0.667**	**0.611**	23	0.89
w/ MonoT5 Filter (40%)	**0.308**	**0.298**	**0.306**	**0.661**	**0.609**	24	0.93
w/ TCT Filter (50%)	0.287	**0.280**	-	0.640	0.599	30	0.94
Doc2Query (n = 80)	0.279	0.267	-	0.627	0.605	30	1.41
w/ ELECTRA Filter (30%)	**0.323**	**0.316**	**0.325**	**0.670**	**0.614**	23	0.95
w/ MonoT5 Filter (40%)	**0.311**	**0.299**	-	**0.669**	**0.612**	28	1.03
w/ TCT Filter (50%)	0.293	**0.283**	-	0.642	0.588	28	1.05

Table 2.4: DOC2QUERY Results, M. Gospodinov and S. MacAvaney and C. Macdonald [10]

The metrics present are: Reciprocal Rank at 10 (RR@10) which is an evaluation metric that measures the quality of a ranked list by considering the position of the first relevant item, assigning the metric higher scores to systems that place relevant items higher in the ranked list and penalizes those that have relevant items placed farther down the list or beyond the top 10 positions; Milliseconds per query (MS/Q) which represents the milliseconds used to generate each query; and lastly GigaBytes (GB) which represents the size in gigabytes of the index.

By taking a look at the results (see Table 2.4), best results occur when using n = 80, which means 80 queries per document, and also when combined with ELECTRA as the filtering neural model. Present ahead from the filters is the value of t which is the value of the threshold named t above, meaning that for example taking a look at the MonoT5 filter, only 40% of the best-ranked queries remain and are used.

In conclusion, a solid technique that leveraged the use of T5, coupled with the use of filtering improved its performance, especially when paired with the ELECTRA neural model.

2.5.1.2 SPLADE

SPLADE, T.Formal et al [7], is a sparse document retrieval method that, like all sparse retrieval methods, is used to improve the first stage of a dual-phase ranking process, document retrieval, featuring a distinctive way of doing document retrieval as a substitute to standard BOW techniques. BOW model is a simple and fundamental technique used in NLP and IR. Although plenty of BOW models exist, problems arise due to the appearance of queries that are not related to the documents, as referenced in DocT5Query , M.Gospodinov and S.MacAvaney and C.Macdonald [10]. However, designing models

for document retrieval is difficult, increasing the demands for techniques that offload a significant portion of the computational burden to an offline phase, enabling rapid online inference. Making these models learn sparse representations to produce queries allows them to inherit characteristics from BOW methods like the efficiency brought by using inverted indexes. This paper, T.Formal et al [7] has, as a basis the SPLADE model and introduces several enhancements that improve its efficiency like a simple modification to SPLADE's pooling mechanism. Also, an extension of the model without query expansion is proposed, enabling offline indexation and distillation techniques to improve its performance. This new version of SPLADE, proposed by this paper, T.Formal et al [7], improves over the last one by using a max pooling strategy, which spawns SPLADE-max, by creating a document-only version called SPLADE-doc and also by combining distillation and the max pooling strategy that is called DistilSPLADE-max.

$$w_j = \sum_{i \in t} \log\left(1 + \text{ReLU}(w_{ij})\right) \qquad w_j = \max_{i \in t} \log\left(1 + \text{ReLU}(w_{ij})\right)$$

(a) Original Representation

(b) Max Pooling operation

In the equations above, w_j represents the weight or value associated with feature j after applying the pooling operation. The set t refers to the terms or tokens in the document or input context that contribute to the computation of w_j. For each feature j, the values w_{ij} are transformed by applying the ReLU (Rectified Linear Unit) function to ensure non-negative values. The logarithmic function is applied to normalize the output, and two different operations are explored: summing over the values in t (see Equation 2.1a) and taking the maximum value (see Equation 2.1b).

SPLADE-max replaces the sum of all the term importance across the tokens (which as a benchmark is done by WordPiece) in the input sequence with a max pooling strategy, as featured in (see Equation 2.1b).

SPLADE-doc is, as mentioned before, a document-only version that eliminates query expansion and query term weighting, increasing performance due to this fact and making the ranking scoring much simpler. Its ranking score is calculated by, with the query q, the document d, and the weight w signifies the importance of token j within the vocabulary concerning token i within the input sequence:

$$s(q, d) = \sum_{j \in q} w_d(j)$$

To perform training for the distillation, the authors concurrently train a SPLADE first-stage retriever and a cross-encoder reranker. In the subsequent step, triplets are generated using a distillation-trained SPLADE. Subsequently, a SPLADE model is trained from the ground up using these newly generated triplets and scores, creating a new variant called DistilSPLADE-max. Results are present in table 2.5.

Model	MS MARCO dev		TREC DL 2019	
	MRR@10	R@1000	NDCG@10	R@1000
SPLADE	0.322	0.955	0.665	0.813
Newer methods				
SPLADE-max	0.340	0.965	0.684	0.851
SPLADE-doc	0.322	0.946	0.667	0.747
DistilSPLADE-max	0.368	0.979	0.729	0.865

Table 2.5: Results of different versions of SPLADE on two datasets, T. Formal et al [7]

2.5.1.3 coCondenser

LM have started to be used to boost dense document retrieval. However, problems arise thanks to their sensitivity to noise in the training data and dependence on large batches for robustly learning the embedding space. CoCodenser, L.Gao and J.Callan [9], is a method proposed to fix these issues having an unsupervised corpus-level contrastive loss to refine the embedding space for passages having as a foundation the Condenser, L.Gao and J.Callan [8], which compresses information into a dense vector through pre-training using an LM. It has a comparative state-of-the-art system named RocketQA (see Table 2.6 and 2.7), Y.Qu et al [28], that enhances the effectiveness of a dense retriever through the implementation of a refined fine-tuning pipeline. This optimized pipeline denoises hard negatives to rectify mislabeling, and large batch training, although with a major problem due to the excessive use of computational resources making it impractical to use.

Method	Batch Size	MRR@10	R@1000
RocketQA			
Cross-batch negatives	8192	33.3	-
+ Hard negatives	4096	26.0	-
+ Denoising	4096	36.4	-
+ Data augmentation	4096	37.0	97.9
coCondenser			
Condenser w/o HN	64	33.8	96.1
+ Hard negatives	64	36.6	97.4
coCondenser w/o HN	64	35.7	97.8
+ Hard negatives	64	**38.2**	**98.4**

Table 2.6: CoCondenser, L. Gao and J. Callan [9], Performance Results compared to RocketQA, Y. Qu et al [28]

The aforementioned issues are addressed through the utilization of the Condenser pre-training architecture. This architecture enables [CLS] vectors to encompass substantial information while exhibiting resistance to noise. [CLS] is a special token added at the beginning of each input sequence used to represent the entire input sequence. This means that [CLS] vectors are vectors generated by [CLS] tokens. In the context of retrieving

from a designated corpus of documents, the training process involves sampling text span pairs from a batch of documents at each step. The model is trained with the objective of ensuring that the [CLS] embeddings of two spans originating from the same document are nearby, whereas spans from distinct documents are positioned far apart. Thus creating a coCondenser, that combines the information-rich [CLS] vectors with the correct [CLS] embeddings.

Method	**MRR@10**	**R@1000**
RocketQA	37.0	97.9
Condenser	36.6	97.4
coCondenser	**38.2**	**98.4**

(a) Performance on MS-MARCO Dev.

Method	**Natural Question Test**			**Trivia QA Test**		
	R@20	**R@5**	**R@100**	**R@5**	**R@20**	**R@100**
RocketQA	74.0	82.7	88.5	-	-	-
Condenser	-	83.2	88.4	-	81.9	86.2
coCondenser	**75.8**	**84.3**	**89.0**	**76.8**	**83.2**	**87.3**

(b) Performance on Natural Questions and Trivia QA Test.

Table 2.7: Comparison between RocketQA and coCondenser in several tests, L. Gao and J. Callan [9]

Concluding, coCondenser allows for a state-of-the-art performance without the need to have a large batch size that allows for the use of fewer resources to train dense retrieval systems.

2.5.2 MonoBERT

MonoBERT, R.Nogueira and K.Cho [22], focuses on adapting BERT to document ranking. Typically, a pipeline of question-answering is focused around 3 stages, with the first being the use of a more general method such as the BM25 (see Subsection 2.1.1). The second consists of a reranking stage, using a more intensive method, generally being a deep learning model. This is attributed to the late appearance of large datasets that, before, were not sufficiently expansive to support neural ranking models like BERT, making them yield irrelevant results. The third phase is an answer generation module that will source its candidate answers from the top ten or fifty documents among these. MonoBERT will only focus on improving the second stage.

The re-ranker's task is to assess the relevance of a candidate passage to a given query by estimating a score. The re-ranking model is based on BERT, making the query input as sentence A, and the passage text serves as sentence B. To adhere to the token limitations, the query is truncated to a maximum of 64 tokens. Additionally, the passage text is truncated to ensure that the combined sequence of the query, passage, and separator

tokens does not exceed 512 tokens. As mentioned before, BERT is still limited when it comes to document size making large documents very difficult to assess whether they are relevant or not.

Utilizing a BERTLARGE model, it was treated as a binary classification model. Specifically, the [CLS] vector is used as input for a single-layer neural network to determine the probability of the passage being relevant or not. This probability is computed independently for each passage, and the final list of passages is obtained by ranking them based on these probabilities.

$$L_{\text{mono}} = -\sum_{j \in J_{\text{pos}}} \log(s_j) - \sum_{j \in J_{\text{neg}}} \log(1 - s_j)$$

In the formula above, J_{pos} is the set of indexes from the relevant passages, and J_{neg} is the set of indexes from the irrelevant passages.Also, s_{j} is the score from the re-ranking of the document j.

Training has led to certain conclusions, including the observation that the model does not necessarily require extensive training, as further training does not significantly improve results. Additionally, it was noted that the model achieved a state-of-the-art performance even with minimal fine-tuning at the time of this paper's publication.

2.5.3 DistilBERT

Transformer models, exemplified by subsection 2.2.2, have dominated the NLP landscape, delivering significant performance improvements. However, with each new iteration, the size of these models tends to escalate, often boasting millions of parameters. Consequently, concerns arise regarding the escalating computational and memory demands, exacerbating environmental impacts and potentially impeding widespread adoption due to the required resources. Consequently, numerous lightweight LM have emerged, characterized by reduced requirements yet achieving comparable performance to their larger counterparts through pretraining with knowledge distillation. Their lighter nature enables pretraining on less robust machines and facilitates fine-tuning processes akin to those of larger models. DistilBERT , V.Sanh et al [35], is a lighter version of BERT slimmed down by the use of knowledge distillation allowing for faster speeds while maintaining 97% of its performance.

2.5.3.1 Knowledge Distillation

Knowledge distillation is a compression method where a compact model, referred to as the student, is trained to emulate the behavior of a larger model, known as the teacher, or an ensemble of models.

In supervised learning, a classification model is typically trained to predict class instances by maximizing the likelihood of correct labels. This entails minimizing the cross-entropy between the model's predicted distribution and the empirical distribution of training labels. A well-performing model will assign high probabilities to the correct class and near-zero probabilities to other classes. However, these probabilities vary in magnitude, reflecting the model's generalization capabilities and its expected performance on unseen data.

The training loss for the student involves a distillation loss computed over the soft target probabilities provided by the teacher. This loss, denoted as Lce, encourages the student to mimic the teacher's distribution, offering a rich training signal. A softmax-temperature parameter, T, was incorporated to control the smoothness of the output distribution. The model score for class i, denoted as zi, is exponentiated and normalized concerning T.

During training, the same temperature parameter, T, is applied to both the student and the teacher, ensuring consistency. However, during inference, T is set to 1 to revert to a standard softmax.

The final training objective combines the distillation loss, Lce, with the supervised training loss. Additionally, a cosine embedding loss, Lcos, is included to align the student and teacher's hidden state vectors, enhancing model convergence and performance.

2.5.3.2 Implementation

The student model adopts a parallel architectural design to BERT. Notably, token-type embeddings and the pooler are omitted, and the layer count is halved. Conclusions were reached that suggest that variations in the hidden size dimension exert a lesser impact on computational efficiency, given a consistent parameter allocation, compared to alterations in factors like layer count. Consequently, the goal was to reduce the layer count to optimize computational efficiency.

Moreover, a critical aspect of the training methodology involves identifying the optimal initialization strategy for the student network to facilitate convergence. Capitalizing on the shared dimensionality between the teacher and student networks, the student is initialized by extracting alternate layers from the teacher. This initialization technique fosters convergence and ensures effective knowledge transfer from the teacher to the student models.

2.5.3.3 Results

After being trained on the same corpus as the BERT model, DistilBERT improves or performs similarly to the ELMo baseline while maintaining 97% of BERT's performance with 40% fewer parameters (see Table 2.8), while also displaying smaller inference time than BERT meaning faster speeds (see Table 2.9).

This paper significantly contributes to the dissertation by presenting a more lightweight variant of BERT, addressing computational constraints associated with larger models. This

Model	Score	COLA	MNLI	MRPC
ELMO	68.7	44.1	68.6	76.6
BERT-base	79.5	56.3	86.7	88.6
DistilBERT	77.0	51.3	82.2	87.5

Table 2.8: Comparison on the dev sets of the GLUE benchmark along with the macro-score (average of individual scores), V. Sanh et al. [35]

Model	# Parameters (Millions)	Inference Time (Seconds)
ELMO	180	895
BERT-base	110	668
DistilBERT	66	410

Table 2.9: Comparison between the number of parameters of each model along with the inference time needed to do a full pass on the STSB development set, V. Sanh et al. [35]

introduction of an alternative model opens avenues for wider applicability in scenarios where computational resources are limited.

2.5.4 PubMedBERT

PubMedBERT paper's, Y.Gu et al [11], scrutinizes the standard practice of using mixed-domain pretraining for domain-specific language models, where models are first trained on general-domain corpora and later fine-tuned with domain-specific data. This methodology assumes that general-domain text aids in the performance of specialized tasks. However, this assumption is contested by this paper's authors, suggesting that pretraining with general-domain data might be unnecessary and potentially harmful for tasks involving specialized domains like biomedical ones.

The primary concern raised is the risk of negative transfer, where the use of general-domain data, which substantially differs in content and context from biomedical texts, could impair the model's performance in the specialized domain. This issue stems from the significant discrepancies between the general-domain and domain-specific data, which may lead the model to learn irrelevant or misleading features that do not translate well to the specialized tasks. For example, a model pretrained on general-domain text might encounter phrases related to daily activities. When later fine-tuned on biomedical data involving terms like 'gene expression' the model could misinterpret key concepts, such as confusing 'binding' in the context of molecules with its more common usage related to books or agreements. This mismatch can result in the model learning irrelevant features, thereby hindering its ability to perform effectively on specialized tasks. These findings underscore the need to reassess the efficacy and applicability of mixed-domain pretraining, especially in fields where abundant and relevant domain-specific data is available.

2.5.4.1 Mixed domain Pre-Training

Mixed-domain pretraining involves starting with a general-domain pretrained model and continuing its training using domain-specific text. For example, BioBERT is initialized with the standard BERT model, pretrained on general-domain text, and then further trained using biomedical texts from PubMed. While this approach is convenient, it has a significant drawback: the vocabulary remains rooted in the general domain, which may not fully represent the specialized biomedical terms. SciBERT is a notable exception that generates its vocabulary from scratch using a mix of biomedical and computer science texts, but it still incorporates out-domain data, which may limit its effectiveness for strictly biomedical applications. The authors of PubMedBERT suggest that pretraining exclusively with in-domain biomedical text may be more advantageous for specific tasks in this domain.

Domain-specific pretraining from scratch offers significant advantages, particularly through the use of an in-domain vocabulary tailored to biomedical texts. General-domain models, even with continual pretraining, often fragment biomedical terms due to an inadequate vocabulary, leading to reduced performance. In contrast, PubMedBERT is trained entirely on biomedical data, effectively captures domain-specific terminology, and avoids the inefficiencies of mixed-domain approaches. This specialized pretraining ensures better optimization and consistently outperforms models that rely on general-domain pretraining.

2.5.4.2 Results

The Hallmarks of Cancer (HoC) corpus, inspired by foundational research on cancer hallmarks, consists of annotated PubMed abstracts with binary labels indicating whether specific cancer hallmarks are discussed. The dataset includes 37 fine-grained hallmarks grouped into ten top-level categories. Initially released with 1,499 abstracts and later expanded to 1,852, this dataset is used for predicting these top-level labels. Unlike previous work that excluded certain abstracts, this study uses the complete dataset and evaluates performance at the abstract level. This dataset is employed to assess PubMedBERT's effectiveness in the document classification task, specifically in identifying discussions related to cancer hallmarks.

The comparison of BERT models applied to biomedical NLP tasks in the HoC benchmark (see Table 2.10) reveals that PubMedBERT, pretrained from scratch using domain-specific data, consistently outperforms other models. These gains are particularly pronounced when compared to models pretrained with out-domain text, such as RoBERTa and the original BERT, which show weaker performance in biomedical tasks. While models like BioBERT, which also uses PubMed data, perform well, PubMedBERT's specialized pretraining strategy, including its in-domain vocabulary, leads to further improvements. Overall, PubMedBERT achieves results that are comparable or superior to previously published benchmarks for BioBERT, SciBERT, and BlueBERT.

Model	BERT uncased	BERT cased	RoBERTa cased
HoC (Micro F1)	80.20	80.12	79.66

Model	BioBERT cased	SciBERT uncased	SciBERT cased
HoC (Micro F1)	81.54	80.66	81.16

Model	ClinicalBERT cased	PubMedBERT uncased
HoC (Micro F1)	80.74	**82.32**

Table 2.10: Performance of different models on the Hallmarks of Cancer (HoC) corpus. The metric used is micro F1.

2.6 TREC 2022 Deep Learning Track

TREC Deep Learning Track is a track that focuses on assessing the performance of ad hoc retrieval techniques within the context of large-scale data environments by performing passage and document ranking. Although this is not the track in which this dissertation will focus and neither passage nor document ranking is needed, this paper is important as document retrieval is performed as an early stage of ranking. It is composed of several models and methods of document retrieval, from sparse to dense document retrieval.

The work presented by Xu et al [42] represents one of the top-performing approaches from a team participating in the aforementioned competition, demonstrating highly competitive results. It used two kinds of representations, that are mentioned before, with BM25, S.Robertson and H.Zaragoza [32], DocT5Query, M.Gospodinov and S.MacAvaney and C.Macdonald [10], and SPLADE, T.Formal et al [7], composing sparse retrieval methods and with CoCondenser, L.Gao and J.Callan [9], being part of the dense retrieval methods.

In this table below (see Table 2.11), methods and their performances are almost exactly in the order that they were mentioned before, with a slight difference. Despite initial expectations, dense retrieval methods, despite their potential for higher computational costs, proved not to be the most suitable choice in this paper's domain. Although this can be due to a dataset where SPLADE can perform better due to certain characteristics, SPLADE reveals itself as the best solo method. To better understand these scores, the NDCG@10 metric is used to evaluate the effectiveness of ranking algorithms.

Moving on to other methods, BM25, S.Robertson and H.Zaragoza [32], performs worse than the other sparse methods thanks to being the basis for both SPLADE, T.Formal et al [7], and Doc2query, M.Gospodinov and S.MacAvaney and C.Macdonald [10]. Doc2Query performs almost similarly to CoCondenser, L.Gao and J.Callan [9], with the former having a slight gain in performance, probably due to its basis being BERT, J. Devlin et al [4]. The best results and also the most important seem to of SPLADE that managed to achieve the higher scores, as evidenced by Table 2.11.

Model	NDCG@10
BM25	0.4370
Doc2query	0.5703
Condenser	0.5778
ROM	0.6115
SPLADE	0.6202
coROM (trained on document)	0.5497
coCondenser (trained on document)	0.5871
coROM (trained on passage)	0.5815
ensemble	**0.7401**

Table 2.11: Ablation experiments result on TREC 2021 document dataset provided by this paper.

Even though the best way to guarantee a good performance in my dissertation would be to ensemble all the methods provided before, it is not possible due to the constraints in computational resources that are available. Thanks to this, SPLADE is the best contribution I can take from this paper.

2.7 TREC Clinical Trials Track

This section will focus on the runs and teams that participated in the TREC Clinical Trials Track 2022, K.Roberts et al [31], to analyze and understand what models were used and what strategies proved to be the most efficient and at the same time relevant to the dissertation.

Clinical trials, crucial for validating medical treatments, are costly, time-consuming, and often struggle to recruit enough eligible patients. Automated methods can enhance the recruitment process by streamlining chart review, swiftly identifying eligible participants, and broadening the pool of potential candidates.

To address the challenges faced in the absence of the Clinical Trials track, an alternative approach was devised, leveraging Electronic Health Records (EHR). In this track, concise descriptions were employed to encapsulate patient profiles, with EHR visit records serving as the document collection or trials. Unfortunately, despite these efforts, IR faced significant obstacles due to insufficient data, leading not only to its ineffectiveness but also to yield unsatisfactory results.

Clinical Trials Track was created to overcome this problem, adopting a patient-to-trials approach and shifting from the traditional trial-to-patients model. This shift allows the creation of an extensive test collection for clinical trial search. In this track, the topic consists of patient descriptions, and the document collection comprises an array of clinical trial descriptions.

Document collection represented clinical trial descriptions, being composed of inclusion and exclusion criteria (see Figures 2.3 and 2.4), which represent what makes a patient

Topic 2 A 32-year-old woman comes to the hospital with vaginal spotting. Her last menstrual period was 10 weeks ago. She has regular menses lasting for 6 days and repeating every 29 days. Medical history is significant for appendectomy and several complicated UTIs. She has multiple male partners, and she is inconsistent with using barrier contraceptives. Vital signs are normal. Serum β-hCG level is 1800 mIU/mL, and a repeat level after 2 days shows an abnormal rise to 2100 mIU/mL. Pelvic ultrasound reveals a thin endometrium with no gestational sac in the uterus.

Figure 2.2: Patient Descriptions , K.Roberts et al [31]

Inclusion Criteria:

1. Written informed consent must be obtained before any assessment is performed.
2. Male and female patients aged 40 - 80 years (inclusive).
3. Diagnosis of knee osteoarthritis
4. Radiographic evidence of tibiofemoral compartment osteoarthritis
5. Pain in the knee during the last 24 hours.The patients should also have had pain in the affected knee on most days over the last month.
6. Patients who are willing to discontinue all non-steroidal anti-inflammatory drugs (NSAIDs) or other analgesic medication taken for any condition, including their knee pain,
7. Patients who are on stable dose of opioids for at least 1 month before screening can continue to take their opioid at this stable dose throughout the study.
8. Patients must also be willing to abstain from any intra-articular or peri-articular injections to the knee or surgery during the treatment period
9. Patients who, if they are currently taking aspirin (325 mg/day or less; as anti-coagulants), are willing to remain on a stable dose one month prior to screening and throughout the study

Figure 2.3: Inclusion Criteria , K.Roberts et al [31]

Exclusion Criteria:

1. Subjects with known hypersensitivity to any biological or investigational drugs.
2. Patients with contraindications to knee injections
3. Patients with joint effusion
4. Patients should not have rheumatoid arthritis or any connective tissue like disease
5. Secondary osteoarthritis with history and/or any evidence of the following diseases: septic arthritis, inflammatory joint disease, gout, Paget's disease of the bone, articular fracture, major dysplasias or congenital abnormality, ochronosis, acromegaly, hemochromatosis, Wilson's disease, primary osteochondromatosis, juvenile chronic arthritis with continued activity in adulthood, heritable disorders (e.g. hypermobility). Patients with secondary osteoarthritis following menisectomy or injuries of a collateral or cruciate ligament are not excluded.
6. Presence or history of underlying metabolic, endocrine, hematologic, pulmonary, cardiac, blood, renal, hepatic, infectious, psychiatric or gastrointestinal conditions
7. Evidence of tuberculosis (TB)
8. One of the risk factors for TB such as:
   (a) Substance abuse (e.g. injection or non-injection)
   (b) Health-care workers with unprotected exposure to patients who are at high risk of TB
   (c) Patients with TB disease before the identification and correct airborne precautions of the patient
   (d) close contact (i.e. share the same air space in a household or other enclosed environment for a prolonged period (days or weeks, not minutes or hours)) with a person with active pulmonary TB disease.
9. Significant medical problems, including but not limited to the following: uncontrolled hypertension,congestive heart failure, uncontrolled diabetes type I and II
10. Subjects with evidence of hepatic or blood coagulation disorders (i.e. hemophilia, etc), anemia, idiopathic thrombocytopenic purpura, or gastrointestinal disorder: severe hepatic disease, history of alcohol and drug abuse; disease of gall bladder and pancreas; active peptic ulceration, gastrointestinal bleeding or history of severe gastro-esophageal reflux disease or severe hiatus hernia; inflammatory bowel disease.
11. Use of any therapeutic protein drug (e.g. anti-tumor necrosis factor alpha (TNFα) antibody)
12. Presence of severe renal function impairment. History of renal trauma, glomerulonephritis, patients with one kidney, or renal failure requiring regular dialysis treatment.
13. Pregnant or nursing (lactating) women, where pregnancy is defined as the state of a female after conception and until the termination of gestation, confirmed by a positive pregnancy test (serum or urine).
14. Subjects with known contra-indications to naproxen (e.g. heart or circulation problems, history of ulcer disease etc.), analgesics, antipyretics, or NSAIDs.
15. Disease of the spine or other lower extremity joints which may interfere with the assessment of the target joint.
16. Surgery on the knee within the last year. Observational arthroscopy, arthroscopic surgery or lavage of the knee within the last 6 months.

Figure 2.4: Exclusion Criteria , K.Roberts et al [31]

eligible and not eligible. This may represent a problem as these criteria can be very long due to the specificity of the trial and use several terms that may not be a direct match to the patient descriptions available.

Topics represent patient descriptions (see Figure 2.2), being composed of several sentences to describe patients and their symptoms. This can also represent a problem due to their length being influenced by having extra information that can be unnecessary to match them to trials.

Matching a patient to a trial can have three categories: **irrelevant** meaning the trial is not relevant, **excluded** where a trial is relevant but the patient complies with exclusion criteria, and **eligible** where the patient is eligible to participate in the trial.

Taking a look at table 2.12 and 2.13, the h2oloo team proves to be the best ranking

Team	NDCG@10	P@10	RPrec
h2oloo	**0.6125**	**0.5080**	**0.3297**
DOSSIER	0.5565	0.4560	0.2810
iiia-unipd	0.5051	0.3980	0.2790
CSIROmed	0.4912	0.3620	0.2136
els_dshs	0.4758	0.3540	0.2128
jbnu	0.4530	0.3220	-

Table 2.12: NDCG@10 + P@10 + RPrec Results, K. Roberts et al [31]

Run Name	Team	MRR
frocchio_monot5_e	h2oloo	0.7262
DOSSIER_2	DOSSIER	0.6607
zs_bert_500	CSIROmed	0.6117
ims_BM25Filtered_kw*	iiia-unipd	0.6085
jbnu2	jbnu	0.5543
phir1m1	phi_lab	0.5516

Table 2.13: MRR Results, K. Roberts et al [31]

team scoring first in all metrics, however, due to a lack of a paper, the strategy used by them cannot be retrieved. The name of the run however points to the use of the Mono variation of T5 which can help identify the model used. The only other team in the top 6 runs that does not provide a paper is DOSSIER, and contrary to the top scoring team, their run name does not imply the model used.

In the upcoming section, a detailed analysis will be carried out on selected participating teams, with a focus on those most relevant to this disertation. Teams without relevant contributions, as previously mentioned, will be excluded. Each team's submission and the runs they executed will be examined to highlight the significance and effort put into their work for this track.

2.7.1 UNIPD

The Unipd team secured the 3rd position across all metrics evaluated (NDCG@10, P@10, and RPrec except in MRR where they obtained the 4th place. Their approach involved employing manual query summaries, implementing query expansion for relevance feedback, and, in alignment with numerous other participating teams in the 2022 edition, incorporating filtering criteria based on gender and age.

2.7.1.1 Query Summarization

This step of the run was done manually, meaning that team members inputted and performed these summaries by themselves instead of using dedicated methods/models. Two types of manual summarization techniques were used, Natural Language Summary (NLS) where ablation operations were used to perform the summarization, and Keyword Summary (KS) where they took the text after NLS summarization and attempted to retain

only keywords based on their termhood, K.Kaguera and B.Umino [13]. Termhood refers to the significance or relevance of terms in representing the content and meaning of a query, making it a method to test what keywords should be displayed. Ablation operations for query summarization involve modifying features, parameters, and data to assess their impact on model performance, meaning only strictly necessary words remain to still form coherent sentences.

2.7.1.2 Query Expansion

To perform query expansion, which enriches the query by adding additional terms to provide a better comprehension, an RM3 , N.A.Jaleel et al [12], V.Lavrenko and W.B.Croft [16], model was used in combination with a BM25 model. RM3 enhances search results by analyzing top-ranked documents, identifying relevant terms, and adding them to the original query. RM3 performs this by ranking the terms to be used, using the probability of the candidate term occurrence. The new query representation is used to re-rank results for improved relevance.

In this context, similar to other teams like , V.Nguyen and M.Rybinski and S.Karim [21], filtering based on gender and age was implemented by downgrading the scores of trials where the patient met the exclusion criteria (age and gender). Rather than removal, these trials were penalized to prevent them from attaining a score high enough for relevance consideration.

2.7.1.3 Results

Measure	Median	(1)	(2)	(3)	(4)	(5)
infNDCG	.392	.410	.446	**.550**	.542	.450
P@10	.258	.300	.200	.300	**.400**	.300
RecipRank	.411	.500	.333	.500	.500	.333

Table 2.14: UNIPD Runs Results, G.M. di Nunzio and G.Faggioli, and S.Marchesin [23]

Run 1 and 3 (see Table 2.14) are similar in using BM25 for every step, leaving out RM3 with the difference between them being using NLS(Run 1) and KS(Run 3). Runs 2 and 4 use the BM25 for retrieval and RM3 for query expansion with the difference between them being the same as runs 1 and 3. 5th run was an experiment, where NLS and BM25 for every step were used, with the addition of the T5 transformer model being used an extra step to the summarization.

As evidenced by the widely known methods of Word2vec and Doc2query, query summarization largely benefits from using keywords only which may explain why KS summarization performs better than NLS. As mentioned before, the RM3 model performs better than only using BM25 thanks to being a dedicated model to generate scores and at

the same time providing query expansion to better determine the relevance of trials to a patient.

2.7.2 CSIROMED

The CSIROMED team paper , V.Nguyen and M.Rybinski and S.Karim [21], focused on the Clinical Trials Track using variations from BERT scoring 4th in NDCG@10 and P@10 and 5th at RPrec. This team has been participating for several years with this year's contribution concentrating on resource-effective self-supervision and using supervision signals from last years judgement. This involved comparing a reranker trained on labeled data from the previous year's track with a self-supervised model trained on the target document corpus. Additionally, the authors explored the efficiency of end-to-end neural ranking, where document representations could be pre-computed, using bi-encoders and neural query expansion. Another experiment that is also relevant was the use of a heuristic to identify patients with the requirements for age/gender that was applied to bi-encoder runs, making them more efficient. Results present in table 2.15.

2.7.2.1 MonoBert Base

This was used in a run called the monobert500, with the first ranking being performed by a BM25, M.Rybinski and J.Xu and S.Karimi [34], with the reranking, as the name suggests, done by a MonoBERT-style reranker.

The MonoBERT-style reranker starts from a SciBERT checkpoint being fine-tuned by previous relevance scores from the previous track, as mentioned before, as well as combining the normalized BM25 score with a softmax function applied to BERT's output with weights being attributed to each method in a 1:9 ratio. The model used is also slightly modified, with the document representations being present in Sentence A and the medical note(query) in Sentence B, making it the opposite of what happens in BERT. This happens as there is no loss in effectiveness, enabling a more direct comparison to a self-supervised run.

2.7.2.2 Self-supervised MonoBERT

The self-supervised MonoBERT which is based on MonoBERT, differs in its reranker model training by training the re-ranker without TREC Clinical Trials Track 2021 data. Instead, it is trained to predict relevance for brief summary inclusion criteria pairs from ClinicalTrials.gov. Positive pairs are from the same document (labeled as 1), while negative pairs are randomly sampled in a 2:1 ratio (labeled as 0).

2.7.2.3 Contrastive Learning with Bi-Encoders

The setup used in the CSIROmedANIR run was based on a SqueezeBERT model with an embedding space initialized by a MetaMap-filtered version of MS Marco. To achieve

this, approximately 100,000 queries containing biomedical entities identified by MetaMap were selected for one epoch of representation learning. The model was further fine-tuned on TREC CT 2021 using a triplet loss function, minimizing the cosine distance between a query and a relevant document while maximizing it from an irrelevant document. SqueezeBERT was chosen based on preliminary results, showing its strength compared to larger language models. Similar to the previous year's approach, the final ranking score is determined using a log-normalized sum incorporating BM25 and cosine similarity between the query and document with it being performed end-to-end on the search engine node without a reranking step. A second run was also made with the intent to rerank the results by sex and age which provided better performance for the trials that had sex and age requirements.

2.7.2.4 Query expansion using DocT5Query trained on MS Marco

The methodology used was described in , R.Pradeep et al [26]. Each topic was enhanced by generating 40 expansions using a pre-trained DocT5query model with each of the original topics and their respective expansions were submitted as queries and assessed using BM25. The outcomes for each query were subsequently aggregated using reciprocal rank fusion. Hyperparameter tuning on the TREC CT 2021 dataset allowed to be employed a fusion approach where the scores were combined, giving a higher weight (in a 20:1 ratio) to the results derived from the original query.

2.8 Final Remarks

The objective of this dissertation, as stated earlier, is to improve upon the results achieved in the Biomarker Literature Retrieval (BLiR) paper (see Subsection 2.4) by further reducing the time required for researchers and experts in the biomedical domain to retrieve relevant papers for the Exposome-Explorer database (see Section 2.3). The goal is to optimize the database curation process by using state-of-the-art document retrieval models.

After reviewing the document retrieval methods available prior to the introduction of cutting-edge transformer models and analyzing various such models, it has been determined that MonoBERT (see Subsection 2.5.2), DistilBERT (see Subsection 2.5.3), PubMedBERT (see Subsection 2.5.4), and BM25 (see Subsection 2.1.1) will be used to accomplish the stated objective.

Method	**Run name**	**NDCG@5**	**NDCG@10**
BM25 w. MonoBERT	monobert500	0.5090	**0.4912**
BM25 w. self-supervised MonoBERT	zs_bert_500	**0.5308**	0.4815
Contrastive Learning with Bi-Encoders	CSIROmedANIR	0.3394	0.3083
Query expansion using DocT5Query	doct5query	0.3626	0.3374
BM25		0.4359	0.4022
TREC Median		**0.3922**	**0.2580**

Method	**Run name**	**P@10**	**RR**
BM25 w. MonoBERT	monobert500	**0.3620**	0.5273
BM25 w. self-supervised MonoBERT	zs_bert_500	0.3280	**0.6117**
Contrastive Learning with Bi-Encoders	CSIROmedANIR	0.2020	0.4085
Query expansion using DocT5Query	doct5query	0.2420	0.3912
BM25		0.2780	0.5150
TREC Median		**0.4114**	

Table 2.15: CSIROMED Runs Results, V. Nguyen and M. Rybinski and S. Karim [21]

3

The Exposome-Explorer Dataset

The dataset used in this research consists of a collection of documents that are highly relevant to biomarkers of exposure to environmental risk factors for various diseases. Each document includes essential information such as the title, abstract, and associated metadata, which provide a comprehensive overview of its content and relevance to the field of biomarker research. This data has been made available as previous work in BLiR, being an earlier version of the Exposome-Explorer database, a key resource for researchers investigating the relationship between environmental exposures and disease risk.

This dataset was initially compiled through the efforts described in the BLiR paper, which developed a retrieval system to identify and collect documents from multiple biomedical sources. Leveraging this pre-existing work, the dataset forms a foundation for investigating key biomarkers related to environmental exposure.

The documents are categorized into three primary types of biomarkers: Dietary, Pollutant, and Reproducibility. Dietary biomarkers are used to trace exposure to nutrients and dietary components; Pollutant biomarkers track exposure to environmental contaminants such as chemicals or toxins; and Reproducibility biomarkers are employed to validate the reliability and consistency of studies related to environmental exposure.

This structured dataset (See Table 3.1) plays a critical role in enabling the analysis and retrieval of documents related to environmental risk factors, helping to enhance the curation process of the Exposome-Explorer database.

Biomarker Type	**Total Documents**	**Total Relevant Documents**	**% Relevant Documents**	**% of Total Dataset**
Pollutant Biomarkers	2,536	247	9.74%	34.85%
Dietary Biomarkers	3,016	156	5.17%	41.45%
Reproducibility Biomarkers	1,718	36	2.10%	23.60%
Total	**7,270**	**439**	**6.04%**	**100%**

Table 3.1: Summary of Biomarker Data

3.1 Dietary Biomarkers

Dietary biomarkers have emerged as valuable indicators of exposure to environmental risk factors associated with the development of various chronic diseases. These biomarkers enable the objective measurement of dietary intake and metabolic responses, providing critical insights into the interactions between diet, environment, and health. The identification and validation of reliable biomarkers have become increasingly important in epidemiological research, where the accurate assessment of exposure is essential for understanding disease etiology and prevention strategies.

In this context, a comprehensive dietary biomarker dataset has been compiled, representing the largest collection of all the type of biomarkers in the Exposome-Explorer database. The dataset consists of 3,016 scientific articles, of which 156 have been identified as relevant to the study of exposure to environmental risk factors. These relevant articles constitute approximately 5.17% of the entire dataset. This extensive dataset serves as a resource for the development of predictive models aimed at uncovering associations between dietary biomarkers and disease risk.

As part of previous work, the Dietary biomarkers dataset has been preprocessed and split into training and testing sets to support the development and validation of machine learning models. Specifically, 70% of the articles (2,111 documents) were allocated to the training set, where 109 relevant articles were identified, accounting for 5.16% of the training subset. The remaining 30% of the dataset (905 documents) was reserved for testing, with 47 relevant articles comprising 5.19% of this subset. This stratified split ensures that both the training and testing sets retain representative distributions of relevant documents, enhancing the robustness of the models developed for biomarker discovery.

"Correlations of vitamin A and E intakes with the plasma concentrations of carotenoids and tocopherols among American men and women", investigates the association between diet and plasma concentrations of carotenoids, retinol, and tocopherols in a sample of 121 men and 186 women. This relevant document title illustrates how the aggregation of the title and abstract provides critical insights into the study's focus, highlighting key associations between nutrient intake and blood plasma concentrations that contribute to the identification of dietary biomarkers.

To identify relevant documents for this research, a predefined set of queries was employed on the World of Science site, available in the annex I.1. By using these keywords, the search strategy effectively retrieved documents that focus on the relationships between dietary intake, biomarker concentrations, and their validation in the context of environmental exposure research.

3.2 Pollutants Biomarkers

Pollutant biomarkers (See Table 3.2) play a critical role in understanding human exposure to environmental toxins and their potential health impacts. These biomarkers are measured in biological matrices such as blood, urine, and serum, providing insights into the levels of hazardous compounds like PAHs, PCBs, HCAs, phthalates, and polybrominated compounds within the human body. The following sections summarize key research areas and datasets related to pollutant biomarkers, highlighting the relevance of each compound in toxicology and environmental health research.

Pollutant Biomarker	**Total Documents**	**Total Relevant Documents**	**% of Relevant Documents**
DPBS	130	7	5.38%
HCAs	120	2	1.67%
PAHs	768	57	7.42%
PCBs	758	63	8.31%
Phthalates	383	37	9.66%
Polybrominated	234	54	23.08%
Polychlorinated	143	27	18.88%
Total	**2,536**	**247**	**9.74%**

Table 3.2: Summary of Pollutant Biomarker Data

DPBS (Dietary, Pollutant, and Biological Systems) biomarkers focus on assessing exposure to dietary and environmental pollutants, providing insights into the interplay between diet, environmental exposure, and health outcomes. The current dataset on DPBS biomarkers includes 130 documents, with 7 identified as relevant (5.38%). These documents contribute to understanding how dietary and environmental pollutants influence biological processes and disease progression.

For Heterocyclic amines (HCAs), biomarkers are used to assess the presence of carcinogenic compounds formed during the high-temperature cooking of meat and fish. The current dataset includes 120 documents, of which only 2 are relevant (1.67%). These studies focus on PhIP adducts as biomarkers for assessing HCA exposure and their potential role in human carcinogenesis, particularly colon cancer.

Polycyclic aromatic hydrocarbons (PAHs) are produced through incomplete combustion processes and are linked to cancer and other diseases. The dataset on PAH biomarkers comprises 768 documents, with 57 identified as relevant (7.42%). These studies often focus on 1-hydroxypyrene (1-OHP) as a biomarker for PAH exposure, providing critical insights into how these pollutants contribute to various health risks, including cancer.

Polychlorinated biphenyls (PCBs) are persistent organic pollutants that accumulate in human tissues and are linked to cancer, immune suppression, and developmental toxicity. The dataset includes 758 documents, with 63 relevant (8.31%). These studies highlight the measurement of PCB isomers in biological samples and their implications for long-term health.

Phthalates, commonly used as plasticizers, are widespread in the environment and raise concerns due to their endocrine-disrupting effects. The dataset includes 383 documents, with 37 relevant (9.66%). Phthalate biomarkers, such as monoester metabolites found in urine, are used to assess human exposure, particularly in vulnerable populations like pregnant women.

Polybrominated compounds, including PBDEs and PBBs, are used as flame retardants and have been associated with reproductive and developmental toxicity. The dataset on polybrominated biomarkers contains 234 documents, with 54 identified as relevant (23.08%). These studies focus on measuring PBDEs and PBBs in human tissues and their potential impact on health outcomes, particularly during prenatal development.

Polychlorinated compounds, such as Polychlorinated Dibenzo-p-Dioxins (PCDDs) and Polychlorinated Dibenzofurans (PCDFs), are toxic pollutants resulting from industrial processes. The dataset on polychlorinated biomarkers includes 143 documents, with 27 identified as relevant (18.88%). These studies assess the bioaccumulation of PCDDs and PCDFs in human tissues and their long-term health risks, including cancer and immune dysfunction.

As detailed in the annex I.2, each type of pollutant biomarker has its own specific queries and keywords tailored to capture relevant studies and documents. These queries are essential in accurately identifying the most pertinent literature related to each class of pollutants and their biological markers. For instance, HCAs are retrieved using terms like "heterocyclic amine," "PhIP," and "adducts," while PAH studies focus on keywords such as "polycyclic aromatic hydrocarbons," "1-hydroxypyrene," and "biological monitoring level."

In total, the pollutant biomarker datasets consist of 2,536 documents, with 247 identified as relevant, accounting for approximately 9.74% of the total documents. These studies form a substantial foundation for understanding the health impacts of environmental pollutants through the use of biomarkers, offering critical insights into human exposure and the associated risks across diverse populations.

3.3 Reproducibility Biomarkers

Reproducibility studies are critical for ensuring the reliability and consistency of studies that investigate environmental exposures and their impact on health. These studies are used to assess the intra- and inter-individual variability in biomarker levels across different individuals and time points, offering a robust framework for validating environmental exposure studies. Reproducibility is essential in epidemiological research to ensure that findings related to biomarkers are consistent and can be generalized to larger populations.

The dataset utilized for the analysis of reproducibility biomarkers is smaller compared to the dietary biomarker dataset. It contains 1,718 documents, with 36 of them identified as relevant to the field of reproducibility. These relevant documents make up approximately 2.10% of the entire dataset. Unlike the more extensive dietary biomarkers dataset, this

smaller dataset was not split into training and testing sets due to its limited size and its use as a benchmark in simpler machine learning methods, as described in previous work by BLiR. This decision reflects the dataset's utility in supporting foundational analyses, but it was not suited for advanced computational modeling due to its scale.

"Evaluation of intra and interindividual variation of urinary 1-hydroxypyrene, a biomarker of exposure to polycyclic aromatic hydrocarbons, explores the variation in urinary 1-hydroxypyrene concentrations, a biomarker of exposure to polycyclic aromatic hydrocarbons (PAHs), among 30 children". This document underscores the role of reproducibility biomarkers in capturing variability in exposure measurements, which is critical for accurately interpreting population-level environmental exposure data.

A set of predefined queries was employed to identify documents relevant to reproducibility biomarkers in the World of Science site, available in annex I.3. The search was designed to capture studies that focus on the repeatability and consistency of biomarker measurements across various biological matrices, such as blood, urine, and adipose tissue. This search strategy enabled the retrieval of key documents that evaluate the reliability of exposure biomarkers in epidemiological research.

3.4 Pre-processing The Data

The methodology for data pre-processing, detailed in the preceding chapter (see Subsection 2.4), established that the 'abstracts+titles' strategy was optimal and yielded superior results across all models. Consequently, this approach was adopted for the entire dataset, which was mostly pre-assembled, necessitating only minor additional processing to meet the specific requirements of the study.

The dataset, accessible via the GitHub repository [15], was organized into three distinct categories based on biomarker types, alongside a comprehensive version that encompassed all biomarker categories and labels. This required the aggregation of all abstracts and titles into a consolidated file, with each line representing a unique combination of an abstract and a title.

In terms of queries, the files came with special characters that were specifically used by the WOS website, most of the time at the end of the keywords. These characters included: *, which allowed for the expansion of search terms by replacing the asterisk with any group of characters, thereby capturing various forms of a word stem to include all possible endings; and $, which was used to indicate that a keyword could appear in either singular or plural forms, necessitating the inclusion of both variations in the search queries. However, to ensure the best possible performance of the BM25 model, it was crucial to standardize the queries by removing these special characters.

The removal of these characters ensured that the queries consisted solely of complete words, aligning them with the BM25 model's design, which is optimized for processing whole words rather than handling expanded or wildcard terms. After removing the *

character, the expanded terms were manually included in their full forms where necessary. Similarly, for the $ character, both singular and plural forms of the keywords were explicitly listed in the queries. To enhance the quality and accuracy of the dataset, duplicate keywords were identified and removed, following the established preprocessing protocol that aimed to eliminate redundancies from the dataset.

Since this dissertation builds upon the foundation established by the previous work conducted in the BLiR paper(see Subsection 2.4), the focus will be primarily on diet biomarkers. These biomarkers were central to the final results presented in the BLiR paper (see Table 2.3), and therefore, they will be emphasized in the analysis presented in this and subsequent chapters. As a result, all reported outcomes in this chapter, as well as in the following ones, will predominantly reflect and highlight the performance and results related to diet biomarkers.

Following the processing of the queries associated with the biomarkers, the dataset was fully prepared and made available for immediate application in this and future modeling efforts.

4

Document retrieval with BM25

This chapter introduces our BM25-based approach to document retrieval for database curation. BM25 is a ranking function extensively utilized in information retrieval, developed from a probabilistic retrieval framework, which enhances the traditional TF-IDF model by incorporating document length normalization and term frequency saturation mechanisms.

This chapter elaborates on the adaptations and implementations of the standard BM25 formula within various experimental contexts. Modifications to term frequency parameters and document length factors are discussed, and designed to optimize performance for particular datasets.

Results from these experiments are subsequently presented, providing a quantitative evaluation of BM25's efficacy on the Exposome-Explorer dataset. These findings are bolstered by performance reviews that highlight the improvements in retrieval effectiveness brought about by the model.

The final discussion assesses the significance of these results, examining the strengths and limitations of the BM25 model. Attention is given to patterns and irregularities observed in the data, providing insights into potential refinements for BM25 in different information retrieval scenarios.

4.1 BM25 package

This section introduces the BM25 package utilized as an initial step in this dissertation to filter out non-relevant documents from the dataset, available at the rank-BM25 package website[3]. The package implements the Okapi (original) BM25 ranking function, along with its variants BM25L and BM25+, enhancing traditional information retrieval strategies as discussed in the seminal paper by Robertson and Zaragoza [33]. By preventing high-frequency terms from overwhelming relevance scores and ensuring that longer documents do not have an undue advantage, the BM25 algorithm optimizes the initial dataset for further analysis. The flexibility to adjust parameters such as k1 and b allows for precise customization to meet the specific needs of the dataset, ensuring that only pertinent

documents are advanced for more detailed processing.

4.1.1 Okapi BM25

The implementation, S.Robertson and H.Zaragoza [32], used by this package is the same as reported in the original paper, meaning no new features are performed over the original.

$$BM25 = \sum_{t \in q} log[\frac{N}{df(t)}] \cdot \frac{(k_1 + 1) \cdot tf(t,d)}{k_1 \cdot [(1-b) + b \cdot \frac{dl(d)}{dl_{xxx}}] + tf(t,d)}$$

In the equation, the symbol t represents a term within the query q, iterating over all terms in the query. N represents the total number of documents available in the document collection or corpus. $df(t)$ signifies the document frequency of term t, indicating how many documents in the collection contain term t. $tf(t,d)$ denotes the term frequency of term t in document d, representing how many times term t appears in document d. dl stands for the length of document d, representing the total number of terms in the document. Lastly, dl_{xxx} denotes the average length of documents across the corpus. BM25 offers flexible customization through two key parameters: k1 and b. These parameters enable users to fine-tune the model to suit the specific properties of their data best. k1 governs the saturation level, with higher values amplifying the saturation effect, meaning it influences how quickly the normalization of term frequencies saturates as term frequencies increase. b, on the other hand, dictates the extent of document length normalization. When b approaches 1, normalization is less pronounced, while values closer to 0 introduce a stronger normalization effect.

4.1.2 BM25L

In a 2011 study presented at SIGIR , Y.Lv and C.Zhai [20], the authors critically examined the document length normalization mechanism

$$\frac{L_d}{L_{avg}} \tag{4.1}$$

in the traditional BM25 ranking function, noting its tendency to favor shorter documents when used in conjunction with cosine ranking. To address this bias, BM25L was developed being an enhancement of the original BM25 formula.

$$rsv_q = \sum_{t \in q} \log\left(\frac{N+1}{df_t + 0.5}\right) \cdot \frac{(k_1 + 1) \cdot tf_{td}}{k_1 \cdot \left((1-b) + b \cdot \left(\frac{L_d}{L_{avg}}\right)\right) + tf_{td}} \tag{4.2}$$

Variables:

- rsv_q: Retrieval status value for the query q, which is the score indicating how relevant a document is to the query.

- $\sum_{t \in q}$: Summation over all terms t in the query q.
- $\log\left(\frac{N+1}{df_t+0.5}\right)$: Logarithm of the inverse document frequency (IDF) term, where:
  - N is the total number of documents in the collection.
  - df_t is the document frequency of term t, representing the number of documents containing term t.
- k_1: A parameter that controls the impact of term frequency on the final score, with larger values allowing term frequency tf_{td} to have a greater influence.
- tf_{td}: The term frequency, or the number of times term t appears in document d.
- b: A parameter for length normalization, typically set between 0 and 1, controlling how document length L_d affects the score.
- L_d: Length of document d.
- L_{avg}: The average document length across the entire document collection.

They re-arrange to get:

$$rsv_q = \sum_{t \in q} \log\left(\frac{N+1}{df_t+0.5}\right) \cdot \frac{(k_1+1) \cdot c_{td}}{k_1 + c_{td}} \tag{4.3}$$

Variables:

- c_{td}: The normalized term frequency, calculated by adjusting tf_{td} based on document length, as shown in the next equation.

where:

$$c_{td} = \frac{tf_{td}}{1 - b + b \cdot \left(\frac{L_d}{L_{avg}}\right)} \tag{4.4}$$

Variables:

- c_{td}: The normalized term frequency, accounting for the length of the document relative to the average document length.
- tf_{td}: The raw term frequency, or the number of times term t appears in document d.
- b: The length normalization parameter, balancing the effect of document length.
- L_d: Length of document d.
- L_{avg}: The average document length across the collection.

The primary modification in BM25L lies in its treatment of the IDF component, where negative values are eliminated, ensuring a non-negative divergence. Further, the authors addressed the disproportionate impact of the length normalization factor (Ld/Lavg) by introducing a positive constant, δ, to this component. The inclusion of δ modifies the normalization curve, shifting its bias towards accommodating longer documents, evidenced by larger Ld values resulting in smaller effective values within the formula.

$$rsv_q = \sum_{t \in q} \log\left(\frac{N+1}{df_t + 0.5}\right) \cdot \frac{(k_1+1)\cdot(c_{td}+\delta)}{k_1 + (c_{td}+\delta)}$$

Evaluations conducted across several datasets by the paper authors demonstrated that BM25L consistently outperformed the standard BM25. The optimal value for δ across these datasets was determined to be 0.5, highlighting its effectiveness in mitigating length-related biases in document ranking. This adaptation allowed BM25 to be adapted to environments characterized by long document lengths.

4.1.3 BM25+

Y.Lv and C.Zhai [20] further identified that the penalization of long documents is not unique to BM25 but also affects other ranking functions. They proposed a comprehensive solution to this issue by setting a lower bound on the impact of a single-term occurrence. This ensures that, irrespective of the document length, a single search term occurrence contributes a minimum predetermined value to the retrieval score.

This methodology diverges from BM25L by incorporating the positive constant δ into the term frequency component (tf_{td}) before its multiplication with the modified IDF term. This modification guarantees that the contribution of a single-term occurrence remains above a certain threshold, even for lengthy documents.

$$rsv_q = \sum_{t \in q} \log\left(\frac{N+1}{df_t}\right) \cdot \left(\frac{(k_1+1)\cdot tf_{td}}{k_1 \cdot \left((1-b) + b \cdot \left(\frac{L_d}{L_{avg}}\right)\right) + tf_{td}} + \delta\right)$$

The authors conducted experiments using several collections, demonstrating that BM25+ consistently outperforms BM25, with a δ value of 1 for optimal performance across several collections.

4.2 Results

In this section, the results of the BM25 model and its variations will be presented, highlighting the necessity of processing each biomarker document type separately to ensure precision and specificity in the search results. Biomarker document types can vary significantly in terms of their characteristics, relevance, and context within the scientific literature. By processing each biomarker document type independently, the search queries could be

tailored to capture the most relevant documents specific to each biomarker document type. This targeted approach reduces the likelihood of retrieving irrelevant studies, thereby enhancing the overall accuracy and quality of the literature review.

A combined run using all biomarkers and queries was also performed but yielded worse results, exacerbated by the inherently broad scope of the queries. When a query encompasses all biomarkers, it becomes excessively large, leading to reduced specificity and increased noise in the search outcomes. This dilution of relevance and precision due to the variability and differing contexts of the biomarkers further underscores the importance of separate processing for each biomarker.

To optimize the retrieval process, a threshold was established where the 20% of documents with the lowest BM25 scores were removed. This threshold was chosen because it was found to effectively balance the retention of all relevant documents while eliminating a significant number of non-relevant ones. By removing the bottom 20% of the documents based on their BM25 scores, the model was able to discard many irrelevant titles without sacrificing any of the relevant content. This approach further refined the search process, ensuring that the remaining documents were of higher relevance and improving the overall quality of the literature review. To optimize the retrieval process, a threshold was established where the 20% of documents with the lowest BM25 scores were removed. This threshold was chosen because it was found to effectively balance the retention of all relevant documents while eliminating a significant number of non-relevant ones. By removing the bottom 20% of the documents based on their BM25 scores, the model was able to discard many irrelevant titles without sacrificing any of the relevant content. This approach further refined the search process, ensuring that the remaining documents were of higher relevance and improving the overall quality of the literature review.

4.2.1 BM25 results

The table 4.1 represents the outcome of running the BM25 model across different threshold values to identify the most effective point for document retrieval and elimination of irrelevant titles. The Success column in the table can have either a Yes or No value, indicating the outcome at each threshold. A Yes in the Success column encapsulates 100% retrieval of all relevant documents, meaning that all relevant files present in the dataset were successfully retrieved. Conversely, a No indicates that not all relevant documents were retrieved, suggesting that the retrieval process missed some relevant files at that particular threshold. The Titles(Docs) Eliminated column shows the number of irrelevant titles that could be discarded while still retaining all relevant documents.

At a threshold of 115.958, the model retrieved 1 relevant document out of 1, achieving a 100% success rate without eliminating any titles. As the threshold decreases, the number of relevant documents retrieved increases, but the success rate begins to vary. When the threshold reaches 35.958, the model starts to eliminate titles while still achieving a 100%

success rate. At this point, 766 titles are eliminated. Upon reaching a threshold of 25.9584, the success rate remains high, with 377 titles eliminated. At a threshold of 23.9584, 206 titles are eliminated.

The optimal threshold, selected for this dissertation, is 21.9584. At this point, 109 relevant documents are retrieved out of the 1971 total documents, achieving a percentage of relevants of 5.53019%. Importantly, 140 titles are eliminated, ensuring that the retrieval process remains efficient while minimizing the loss of relevant information. The threshold of 21.9584 was chosen for this dissertation to balance retaining all relevant documents while eliminating a significant number of irrelevant titles. This approach ensures coherence across all biomarkers, as each can successfully remove the documents with the lowest 20% scores according to the model.

Top Value	**Number of Relevant Files**	**Percentage of Relevant Files**	**Total Number of Files**	**Success**	**Titles Eliminated**
115.958	1	100.0	1	No	-
95.958	13	61.9048	21	No	-
75.958	40	39.2157	102	No	-
55.958	83	17.7733	467	No	-
35.958	109	8.1049	1345	Yes	766
25.9584	109	6.28604	1734	Yes	377
23.9584	109	5.72178	1905	Yes	206
21.9584	**109**	**5.53019**	**1971**	**Yes**	**140**
19.9584	109	5.4148	2013	Yes	98
15.958	109	5.24038	2080	Yes	31
-0.0415642	109	5.16343	2111	Yes	0

Table 4.1: Results achieved using OKAPI BM25 in the Diet biomarkers

Top Value	**Number of Relevant Files**	**Percentage of Relevant Files**	**Total Number of Files**	**Success**	**Titles Eliminated**
157.233	0	0.0	1	No	-
137.233	1	33.3333	3	No	-
117.233	1	25.0	4	No	-
97.2333	2	16.6667	12	No	-
77.2333	10	20.8333	48	No	-
57.2333	27	17.1975	157	No	-
37.2333	32	7.15884	447	No	-
23.2333	**36**	**3.71134**	**970**	**Yes**	**747**
21.2333	36	3.36449	1070	Yes	647
19.2333	36	3.04054	1184	Yes	533
-0.766687	36	2.09668	1717	Yes	0

Table 4.2: Results achieved using OKAPI BM25 in the Reproducibility biomarkers

The comparison of the two tables, using the same model in different biomarkers,

highlights the effectiveness of the 20% threshold for document retrieval using the BM25 model. In the first table (see Table 4.1), the optimal threshold of 21.9584% balances retaining relevant documents and eliminating irrelevant titles, with 140 titles removed while maintaining a high success rate. Similarly, the second table (see Table 4.2), with a threshold of 23.2333%, achieves a successful retrieval of all relevant documents while eliminating 747 titles. Both tables demonstrate that the 20% threshold across all biomarkers consistently filters out irrelevant documents, ensuring efficient and precise retrieval across different datasets.

4.2.2 BM25L results

The table 4.3 shows the results achieved on the diet biomarker, using the BM25L variant of the BM25 model. While comparing both models, several key differences emerge, particularly in threshold values, score variability, and document removal efficiency.

Firstly, the threshold values for BM25 are significantly lower than those for BM25L. For instance, the top value for BM25 is 115.958, whereas for BM25L it is 554.8. This substantial difference indicates that BM25L assigns higher scores to documents, likely due to a more aggressive scoring mechanism that benefits larger documents.

Secondly, the range of values in the BM25 model is narrower compared to the BM25L. This suggests that BM25L has a broader spread of scores, reflecting greater diversity in document relevance. The wider variability in BM25L scores may be attributed to its sensitivity to specific document features, such as size.

Lastly, the BM25 model outperforms BM25L in terms of document removal efficiency. At an optimal threshold of 21.9584, BM25 eliminates 140 titles while retaining all relevant documents. In contrast, BM25L at its optimal threshold (20.7997) removes only 51 titles. This demonstrates BM25's superior ability to filter out irrelevant documents, making it more effective in reducing noise and enhancing the quality of retrieved documents.

Top Value	**Number of Relevant Files**	**Percentage of Relevant Files**	**Total Number of Files**	**Success**	**Titles Eliminated**
554.8	0	0.0	1	No	-
472.8	1	50.0	2	No	-
374.8	3	25.0	12	No	-
252.8	18	28.125	64	No	-
202.8	34	19.1011	178	No	-
124.8	85	13.3229	638	No	-
64.7997	107	7.05805	1516	No	-
24.7997	109	5.3379	2042	Yes	68
20.7997	**109**	**5.29383**	**2059**	**Yes**	**51**
2.79965	109	5.16833	2109	Yes	1

Table 4.3: Results achieved using BM25L in the Diet biomarker

4.2.3 BM25+ results

The table 4.4 shows the results achieved on the diet biomarker, using the BM25+ variant of the BM25 model. The comparison between both models reveals significant differences in their scoring mechanisms, document score ranges, and efficiency in document elimination.

Firstly, the threshold values in the BM25 model are substantially lower compared to those in the BM25+ model. For instance, the top value for BM25 is 115.958, whereas for BM25+, it is 1046.76. This considerable difference suggests that the BM25+ model assigns much higher scores to documents, potentially due to a different scaling factor or a more aggressive scoring mechanism inherent to the BM25+ algorithm. This is explained by the fact that by adding δ to the term frequency component, BM25+ increases the overall retrieval score for documents that contain the search terms, which ensures a minimum contribution from each term occurrence and reduces the penalization of long documents.

Moreover, the values in the BM25 model exhibit less variation compared to those in the BM25+ model. The BM25 model's top ten threshold values range from 115.958 to -0.0415642, while the BM25+ model's top ten threshold values range from 1046.76 to 944.76. This indicates that the BM25+ model has a wider spread of scores, reflecting a greater diversity in document relevance as assessed by the model. Such variation could be due to BM25+'s enhanced sensitivity to specific document features.

In terms of document removal efficiency, the BM25 model demonstrates superior performance. At the optimal threshold of 21.9584, the BM25 model successfully eliminates 140 titles while retaining all relevant documents. This achievement highlights the model's effectiveness in filtering out irrelevant documents while preserving the relevant ones, a crucial factor in practical applications where reducing noise in retrieved documents is essential. The BM25+ model, even at its optimal threshold of 966.76, manages to remove only 68 titles. This suggests a potential trade-off between precision and recall within the two models. Precision appears to be more finely tuned in the BM25 model. It manages to minimize false positives more effectively, ensuring a cleaner dataset without losing pertinent information. However, the recall remains high for both models as they successfully retain all necessary titles.

In summary, while BM25+ offers higher and more varied scores, indicating a different approach to relevance assessment, BM25 provides more efficient document elimination. These differences highlight the trade-offs between the two models, with BM25+ potentially offering better differentiation among documents but BM25 achieving better efficiency in reducing irrelevant documents.

BM25L and BM25+ are both variants of the BM25 model, and a comparison of their performance on the diet biomarker dataset reveals several key differences in their scoring mechanisms, threshold values, and document elimination efficiency.

Firstly, threshold values differ substantially between the two models. BM25L assigns lower scores to documents, with its top threshold value being 554.8, whereas BM25+ assigns much higher scores, with a top value of 1046.76. This difference suggests that

Top Value	Number of Relevant Files	Percentage of Relevant Files	Total Number of Files	Success	Titles Eliminated
1046.76	1	0.0	1	No	-
1036.76	3	50.0	2	No	-
1026.76	12	25.0	12	No	-
1016.76	21	28.125	64	No	-
1006.76	40	19.1011	178	No	-
996.76	60	13.3229	638	No	-
986.76	83	7.05805	1516	No	-
976.76	**99**	**5.3379**	**2042**	**Yes**	**68**
966.76	108	5.29383	2059	Yes	51
944.76	109	5.16343	2112	Yes	0

Table 4.4: Results achieved using BM25+ in the Diet biomarker

BM25+ has a more aggressive scoring mechanism, likely due to the addition of an δ term to the term frequency component, which boosts the retrieval score for documents with search terms, particularly benefiting longer documents. In contrast, BM25L places greater emphasis on normalizing document length, which leads to lower and more compressed scores.

Secondly, score variability is another notable distinction. BM25+ exhibits a wider spread of scores, ranging from 1046.76 to 944.76 in its top thresholds, while BM25L's scores range from 554.8 to 2.79965. The broader spread of BM25+ indicates greater sensitivity to document relevance and more differentiation among documents. This wide variability may reflect BM25+'s ability to account for a range of document features more effectively. BM25L, with its narrower range of scores, suggests a focus on adjusting for document length while maintaining tighter control over the distribution of relevance scores.

In terms of document elimination efficiency, BM25L performs slightly better. At its optimal threshold of 20.7997, BM25L eliminates 51 titles while retaining all relevant documents. On the other hand, BM25+ at its optimal threshold of 976.76 eliminates 68 titles, slightly more than BM25L. However, the difference in document elimination is marginal, and BM25L's superior performance in this aspect indicates that it may be more effective in filtering out irrelevant documents while still retaining all relevant ones. This makes BM25L more efficient at reducing noise in the retrieved documents, which is crucial for ensuring higher precision.

Despite BM25+'s slightly lower document elimination efficiency, its relevance assessment is particularly strong. By adding δ to the term frequency, BM25+ reduces the penalization of longer documents, making it more effective in datasets where longer documents play a crucial role. This scoring modification helps ensure that longer documents are not unfairly ranked lower, thereby enhancing recall. BM25L, by focusing on document length normalization, achieves better precision, particularly in datasets with a mix of short

and long documents, by avoiding over-rewarding longer texts.

In conclusion, BM25+ excels in providing higher and more differentiated relevance scores, especially benefiting longer documents due to its scoring enhancements. BM25L, however, outperforms BM25+ in terms of document elimination efficiency, making it more suitable for situations where reducing irrelevant documents is critical to improving precision. Both models retain high recall, but their trade-offs between precision, score variability, and document elimination efficiency make each model more suited to specific types of datasets.

4.3 BM25 Application

After applying the model to dietary biomarkers, a thorough process of data refinement was conducted specifically on dietary biomarkers documents. This involved carefully filtering the dataset by removing low-scoring articles to improve the quality of the data used for analysis. For this analysis, the threshold for each biomarker was deliberately set to the lowest 20% of scores, as previously mentioned. The dataset, already split into training and test sets, saw the exclusion of 140 articles from the training set and 95 from the test set, reducing the total from 3016 articles to 2781, which effectively cut the data pool by approximately 7.79%. The choice of this threshold was strategically selected to balance the preservation of all pertinent documents while removing a substantial number of irrelevant titles. This approach ensures consistency across all biomarkers, as each can effectively eliminate the documents that fall within the lowest 20% of scores as determined by the model, ensuring a streamlined and focused dataset for further analysis.

On reproducibility biomarkers, a similar approach was followed that exhibited a more pronounced reduction in data volume. By applying the previously mentioned threshold of the lowest 20% of scores, a substantial number of 647 articles were discarded, reducing the dataset from 1717 articles to 1070. This accounted for about 37.6% of the initial article pool. The substantial reduction of data emphasizes the focused effort to distill the most impactful and pertinent information from a broad dataset, which is critical in achieving the objective and ensuring the integrity and accuracy of its conclusions.

However, when addressing pollutant biomarkers later in the study, the limitations of the BM25 model became apparent. Given the smaller size of the relevant document pool, the model frequently found no grounds for document removal, resulting in multiple instances where no articles were excluded. This issue highlighted the challenges of applying the same model across different sections with varying data volumes, impacting the model's effectiveness in filtering and refining the dataset for precise analysis.

4.4 Final Thoughts

The BM25 model played an important role in refining the datasets for this dissertation, proving instrumental in creating a streamlined and more focused body of research articles.

By effectively eliminating less relevant documents through the application of a threshold based on the lowest 20% of scores, the model facilitated a significant reduction of noise within the datasets associated with dietary and reproducibility biomarkers. This cleaning process was essential for maintaining a high-quality dataset that could be further utilized in deep learning models.

Despite its straightforward approach to document ranking, the BM25 model demonstrated substantial efficiency, with the potential to be integrated with more complex methodologies, such as deep learning models. The objective is to combine BM25 with a deep learning model to create a robust pipeline for document retrieval. This integrated approach aims to leverage the precision of BM25 in ranking and filtering relevant documents while enhancing the adaptability and analytical depth provided by deep learning techniques. Such a pipeline can be particularly valuable in biomarker studies, ensuring that the models not only retain their effectiveness in filtering relevant information but also adapt to the nuanced demands of varied biomarker research. By incorporating BM25 as a foundational step, the overall efficiency of the research pipeline can be significantly improved, facilitating the retrieval of highly relevant documents in a biomedical domain.

In conclusion, while the BM25 model faced limitations in handling smaller datasets, such as those related to pollutant biomarkers, its overall contribution to this dissertation underscores its value. The ability to reduce dataset size while maintaining integrity and relevance suggests that even simpler ranking algorithms like BM25 can be not only effective on their own but also serve as valuable components in more sophisticated data processing pipelines, potentially enhancing the capabilities of deep learning models to deliver more accurate and reliable results.

5

BERT Based Document Retrieval Models

The advent of BERT has marked a significant milestone in the field of NLP. Introduced, J.Devlin et al [4], in 2019, BERT has revolutionized the way machines understand and interpret human language. Its ability to capture the context of words in a sentence, both from the left and the right, has set it apart from previous models that relied on unidirectional context. This breakthrough has made BERT a cornerstone in various NLP applications, including document retrieval, question answering, and sentiment analysis.

BERT's importance lies in its deep bidirectional nature, which allows it to understand the intricacies of language more effectively than its predecessors. By pre-training on a vast corpus of text and fine-tuning for specific tasks, BERT achieves state-of-the-art performance in a wide array of benchmarks. This versatility and robustness have led to the development of numerous derivative models that build on BERT's architecture to address specific needs and domains more effectively. These derivative models often retain the core advantages of BERT while introducing modifications to enhance performance, efficiency, or domain-specific understanding, which in the case of this dissertation needs to be of the biomedical domain.

This chapter delves into several BERT-based models that have been utilized in the context of this dissertation, specifically focusing on MonoBERT , R.Nogueira and K.Cho [22], DistilBERT , V.Sanh et al [35], and PubMedBERT, Y.Gu et al [11]. Each of these models represents a unique adaptation of the original BERT framework, tailored to meet distinct requirements.

MonoBERT, R.Nogueira and K.Cho [22], as discussed previously, is fine-tuned specifically for document retrieval tasks. Its ability to rank documents based on relevance has proven highly effective, especially in the biomedical domain. This model's success in competitions like the TREC Clinical Trials Track 2022 (see Subsection 2.7.2) highlights its potential for improving information retrieval systems.

DistilBERT, V.Sanh et al [35], on the other hand, aims to provide a lighter, faster version of BERT without significantly compromising performance. By distilling the knowledge

from BERT into a smaller model, DistilBERT reduces the computational resources required, making it suitable for environments where efficiency is critical.

PubMedBERT, Y.Gu et al [11], is another specialized variant, pre-trained exclusively on biomedical literature. This model leverages the rich, domain-specific language found in biomedical texts to offer superior performance in tasks such as medical document classification and retrieval. By focusing on a corpus relevant to the biomedical field, PubMedBERT ensures a deeper understanding of the terminology and context unique to this domain.

This chapter explores the utilization of these BERT-based models in this dissertation. Their diverse adaptations underscore the flexibility and enduring impact of the original BERT architecture. Each model's unique strengths contribute to advancing the capabilities of NLP applications within the biomedical domain, aligning with the objectives of this dissertation to enhance document retrieval systems.

5.1 MonoBERT

As mentioned earlier, the field of NLP has seen tremendous advancements with the development of transformer-based models with BERT emerging as a pivotal model due to its ability to understand the context of words in a sentence more effectively than its predecessors. However, BERT's utility extends beyond basic NLP tasks, finding significant applications in more specialized areas, such as document retrieval. In this context, MonoBERT, R.Nogueira and K.Cho [22], represents an important adaptation of the BERT architecture, tailored specifically for improving information retrieval tasks. MonoBERT is a fine-tuned version of BERT, optimized for document retrieval. While BERT was originally designed for a variety of NLP tasks like question answering and language inference, MonoBERT specifically enhances BERT's ability to rank documents. It achieves this by training on large-scale datasets where the model learns to score documents based on their relevance to a given query. This fine-tuning process involves the model learning the nuances of matching queries to relevant documents, making it particularly effective in contexts where precision in retrieval is paramount.

The biomedical domain presents unique challenges for document retrieval due to the specialized and complex nature of the terminology and the critical importance of precise information. In this dissertation, which focuses on document retrieval in this domain, utilizing a model like MonoBERT can be highly advantageous. The model's ability to understand and rank relevant documents more accurately can significantly enhance the retrieval process, ensuring that pertinent biomedical literature is retrieved effectively. The Text Retrieval Conference (TREC) Clinical Trials Track 2022, K.Roberts et al [31], is a competition that evaluates the performance of document retrieval systems in the context of clinical trials. This competition involves systems attempting to match patients with relevant clinical trials based on detailed eligibility criteria. Participants are provided with

a set of clinical trial descriptions and patient cases, and their task is to rank the trials according to their relevance to each patient case.

MonoBERT's impressive TREC Clinical Trials Track 2022 performance underscores its capabilities. Being one of the top-performing models in this competition demonstrates its efficacy in handling the intricate requirements of biomedical document retrieval. Its success is attributed to its refined ability to parse and understand complex medical terminology and to rank documents with high relevance accurately. With MonoBERT's fine-tuned architecture for ranking documents, the retrieval system can achieve higher precision, ensuring that the most relevant and critical information is retrieved. The model's adaptation to handle complex biomedical language and its proven track record make it an ideal choice for research focus.

5.1.1 Adapting MonoBERT to the Dataset

MonoBERT was the first model to be used in this dissertation, applying a zero-shot learning approach to evaluate the relevance of documents to specific queries as the initial step. Zero-shot learning is a paradigm in machine learning where a model is tasked with making predictions for classes or data points it has not been explicitly trained on. This approach leverages the model's ability to generalize from known contexts to new, unseen situations, thus enabling it to perform tasks without direct prior exposure to specific instances. In the context of document relevance scoring, a zero-shot approach involves using a pre-trained model to assess the relevance of documents to given queries without the need for further fine-tuning on a specific dataset.

5.1.1.1 Maximum Token Size

MonoBERT, as well as other BERT-based models, has a token input limitation of approximately 512 tokens, which impacts its ability to process longer texts in a single pass. This restriction means that any text longer than 512 tokens must be truncated or split into smaller chunks, potentially leading to loss of context and incomplete analysis. This limitation can affect tasks requiring a comprehensive understanding of lengthy documents, such as document classification, and information retrieval. This dissertation employed a strategy to split texts into chunks, so that all the tokens of the document can be used by the model, and then calculate the average of all the chunks, ensuring that the model could handle longer texts while preserving as much contextual information as possible, something that will be mentioned in the next subsection (see Subsubsection 5.1.1.2).

5.1.1.2 Evaluation Setup

To further understand the limitations and capabilities of MonoBERT in this context, a specific testing script was developed and utilized. This script played a critical role in systematically evaluating the model's performance on document relevance tasks.

The purpose of this script is to evaluate the relevance of a collection of documents concerning a set of predefined queries. This is achieved through tokenization of the documents and queries, preparation of inputs for a machine learning model, and computation of relevance scores via model inference. The ultimate goal is to derive average relevance scores for each document, indicating its pertinence to the queries.

The process begins by loading the document paths and creating two numpy arrays: one containing the concatenated abstracts and titles of the documents and another containing the queries. Each document within the article list is tokenized using a pre-trained tokenizer from the HuggingFace Transformers library. Tokenization involves converting the text into tokens that the model can understand by mapping words or subword units into numeric representations based on the tokenizer's vocabulary. It also includes adding special tokens as necessary and ensuring a consistent format through truncation and padding.

Similarly, each query from the queries list undergoes tokenization, resulting in a list of tokenized queries. This ensures that both documents and queries are in a compatible format for subsequent comparison.

The tensors from the queries, along with those of the document, are concatenated to create a combined input that encapsulates the relationship between the document and the query. However, due to a limitation in the model's maximum input length (512 tokens), the combined input had to be split into chunks. This led to a problem where the query and document inputs were poorly placed within these chunks. As a result, the query would either only appear fully in the first chunk, or be split into parts across multiple chunks, with portions of both the query and the document dispersed between them. This poor placement hindered the model's ability to fully capture the relationship between the document and the query, as parts of the input were not optimally organized.

For each chunk of the combined input, the model performs inference without gradient calculation, conserving memory and computational resources since the model is not being trained. The model's output logits are used to derive a relevance score for each chunk, aggregated into a list.

The average score for the query against the document is computed by averaging the chunk scores and adding this average to the total query score. Once all queries for the document are processed, the average document score is calculated by dividing the total query score by the number of queries, and this score is appended to the document scores list. The loop counter is incremented, and the process repeats for the next document until all documents in article list are evaluated.

After obtaining the relevance scores, the script classifies the documents based on a threshold value of 0.5. If the score is equal to or greater than the threshold, the document is classified as relevant (class 1); otherwise, it is classified as not relevant (class 0).

Several evaluation metrics are calculated, including precision, recall, F1-score, F2-score, and ROC AUC. These metrics provide a comprehensive assessment of the model's performance in terms of classification accuracy and the ability to distinguish between

relevant and non-relevant documents. These metrics are used in this dissertation as they are the ones used in the BLiR paper, A.Lamurias et al [14], making them the comparison points between models and results.

5.1.2 MonoBERT's Results

Initially the results of the zero-shot approach were much lower than expected, leading to a detailed examination of the model's implementation. It was discovered that the original MonoBERT paper, R.Nogueira and K.Cho [22], implementation employs a softmax layer to compute scores, a step that was absent in this dissertation's initial implementation.

To address this discrepancy, a softmax layer with two labels was incorporated to accurately compute the scores. This modification ensures that the model aligns with the methodology presented in the original MonoBERT paper, thereby improving the reliability and accuracy of the results.

Metric	**Softmax**	**Zero-shot**
ROC AUC	0.6034	0.4910
F2-score	0.0996	0.0893
F1-score	0.0436	0.0393
Recall	0.6944	0.5833
Precision	0.0225	0.0203

Table 5.1: MonoBERT's result for Zero-shot and Softmax Runs

The initial results (see table 5.1 on the Zero-shot collumn) show a high recall but significantly low precision, indicating that the model identified many relevant instances but produced numerous false positives. The F1-score and F2-score, which balance precision and recall, were correspondingly low, reflecting this trade-off. Additionally, the ROC AUC of 0.491 suggests that the model's ability to distinguish between positive and negative classes is only marginally better than random guessing.

Several metrics improved noticeably with the addition of a softmax layer (see table 5.1 on the Softmax collumn). Precision increased slightly to 0.0225, and recall improved significantly to 0.6944, indicating an enhanced ability to identify relevant instances with a marginal reduction in false positives compared to the zero-shot approach. As a result, both the F1-score and F2-score showed improvement, demonstrating a better balance between precision and recall. Notably, the ROC AUC increased to 0.6034, suggesting a more reliable distinction between positive and negative classes.

However, MonoBERT was not used more extensively in this dissertation due to changes in its availability from the preparation phase to the start of the practical component. Only the large version of MonoBERT was available, which made running and training very slow and resource-demanding. The large version has significantly more parameters, leading to increased computational requirements for both memory and processing power. This results in longer processing times and higher demands on hardware resources.

This application aimed to demonstrate the model's ability to generalize from its pre-trained knowledge base. However, the results indicated that MonoBERT's performance was not optimal for this task. Despite this, utilizing MonoBERT in this manner served as a crucial first step in the search for the best model for document retrieval in the biomedical domain. This initial attempt aimed to observe the model's performance without training, highlighting the challenges. This approach ultimately contributed to the advancement of more effective models for document relevance scoring by providing a baseline for comparison.

As a consequence of the increased computational resources, only a zero-shot approach and improvements to the testing script were implemented. This limitation highlights how much computational resources are needed for deep learning models, especially when working with large-scale models like MonoBERT.

5.2 DistilBERT

DistilBERT is a lighter version of BERT, created through a process known as distillation, which reduces the model size while retaining much of the original model's performance. DistilBERT achieves this by training a smaller student model to mimic the behavior of the larger BERT model, effectively capturing its knowledge in a more compact form.

DistilBERT was chosen for application in this dissertation due to its ability to provide a powerful context understanding of BERT while being more resource-efficient. This decision was also influenced by the challenges encountered with MonoBERT, whose large size and computational demands proved to be a significant limitation with the available resources. The efficiency of DistilBERT makes it particularly advantageous in scenarios where computational resources are limited, or where the speed of model inference is critical. The constraints on available computational resources strongly influenced the choice to use DistilBERT. Despite its reduced size, DistilBERT has been shown to perform on par with BERT across various NLP tasks, thanks to its carefully engineered training process. Additionally, as a more recent development, DistilBERT benefits from improvements and optimizations made after BERT's initial release, such as compression techniques and optimization strategies developed after BERT's initial release, further enhancing its performance and applicability.

In the context of document retrieval, particularly within the biomedical domain, DistilBERT offers a compelling balance between accuracy and efficiency. The biomedical field often requires processing large volumes of data quickly and accurately, and the ability of DistilBERT to handle complex language tasks faster makes it a valuable tool in this domain. By incorporating DistilBERT into the document retrieval pipeline, this dissertation aimed to leverage these advantages to enhance retrieval effectiveness while reducing computational overhead.

The application of DistilBERT within this dissertation not only seeks to improve the efficiency of the retrieval system but also aims to maintain, if not exceed, the precision

offered by larger models like BERT. This dual focus on performance and efficiency aligns with the overarching goals of the research, ensuring that the retrieval system can deliver high-quality results in a practical and resource-conscious manner. DistilBERT's proven ability to balance these requirements, combined with the limitations of available resources and the issues related to MonoBERT's size, makes it a strategic choice for advancing the document retrieval objectives outlined in this work.

5.2.1 DistilBERT's Training

In the initial phase of the research, the focus was on leveraging a zero-shot learning approach using the MonoBERT model (see Subsection 5.1.2), which allowed for applying pre-trained models without further fine-tuning. However, this approach was limited by the lack of access to sufficient computational resources, which precluded the possibility of training or fine-tuning models on domain-specific data. However, the resources available were enough to fine-tune a DistilBERT model.

Parameter	**Description**
`num_train_epochs`	The number of epochs for training.
`per_device_train_batch_size`	The batch size for training on each device.
`per_device_eval_batch_size`	The batch size for evaluation on each device.
`warmup_steps`	Number of steps for learning rate warmup.
`weight_decay`	Weight decay factor to prevent overfitting.
`learning_rate`	Learning rate used for training.
`seed`	Seed for reproducibility of results.

Table 5.2: Training parameters used for optimizing BERT-Based models.

The DistilBERT base uncased variant was the variant of the DistilBERT model used, with the model and its corresponding tokenizer being initialized using the Hugging Face transformers library.

The dataset for training and evaluation is constructed by first loading and preparing the text data (article_list) and corresponding labels (true_labels). These are then converted into a NumPy array format for ease of manipulation.

A custom PyTorch dataset class, CustomDataset, is defined to handle the tokenization and encoding of the input texts. The tokenizer processes each text entry, applying padding and truncation to ensure a consistent input length of 512 tokens. The dataset class also manages the alignment of text inputs with their respective labels, handling potential indexing errors with an exception mechanism, thanks to the variety of data from each biomarker.

The dataset is subsequently split into training and validation subsets, with 80% of the data allocated to training and the remaining 20% reserved for validation. This split ensures that the model is trained on a comprehensive portion of the data while retaining a separate set for performance evaluation.

To ensure the reproducibility of results, a fixed seed value is set across the different libraries used. This step guarantees that the random processes involved in training, such as data shuffling and weight initialization, produce consistent outcomes across different runs of the experiment, allowing for the replication of the procedures performed in this dissertation.

After providing a basis for the script, the next logical step was to select hyperparameters (see Table 5.2 for further insights on each hyperparameter) to fine-tune and train this model, based on several papers mentioned in the Chapter 2.

Parameter	**Value**
num_train_epochs	3
per_device_train_batch_size	8
warmup_steps	500
weight_decay	0.01
evaluation_strategy	epoch

Table 5.3: First run's parameters used for training DistilBERT

In the early stages of experimentation with DistilBERT, the initial training parameters (see Table 5.3), marked the first substantial evaluation of model performance. The results from this evaluation are provided in the table 5.4, with the run being called Run 1. These metrics provided a baseline assessment of the model's effectiveness.

Run Name	**Precision**	**Recall**	**F1-score**	**F2-score**	**ROC AUC**
Run 1	0.5581	**0.5106**	0.5333	0.5195	**0.9361**
Run 2	**0.6486**	**0.5106**	**0.5714**	**0.5333**	0.9146
Run 3	0.6154	0.3404	0.4384	0.3738	0.9092

Table 5.4: Results of DistilBERT's runs

A second round of training was conducted using these refined hyperparameters (see Table 5.5), focusing on fine-tuning. These refined hyperparameters were acquired after conducting several experiments to identify the optimal values that would improve the model's performance. Several hyperparameters were added and changed during this phase, leading to significant modifications to the training script. One critical hyperparameter was the learning rate, set at 1e-5, which controls the speed at which the model learns by dictating the impact of the updates made to its weights during training.This value was chosen because it is widely used in various studies, offering a balance between fast convergence and the avoidance of overshooting the optimal solution. This rate has been proven

to provide stable convergence and robust performance across different architectures and datasets, making it a reliable choice. Furthermore, empirical testing has confirmed that this rate not only ensures stable convergence and robust performance across various architectures and datasets but also consistently outperforms alternative values, establishing it as a more effective choice for achieving reliable results.These changes resulted in a refined set of parameters that would be consistently applied throughout the remainder of this dissertation.

The increase in the number of training epochs, coupled with the addition of updated hyperparameters as shown in the corresponding table, contributed to an improvement in model performance. The evaluation of this training phase is shown in the table 5.4 in the Run 2 row. These metrics reflect the positive impact of the refined training strategy and the careful selection of a widely recognized learning rate on the model's ability to accurately classify and predict outcomes, demonstrating a substantial improvement over the initial evaluation.

Parameter	**Value**
num_train_epochs	10
per_device_train_batch_size	8
warmup_steps	500
weight_decay	0.01
eval_strategy	epoch
save_strategy	epoch
learning_rate	3e-5

Table 5.5: Second run's parameters used for training DistilBERT

An additional experiment was conducted to increase the batch size from 8 to 16. Contrary to expectations, this adjustment led to a decline in performance, indicating that the model's efficiency did not scale linearly with batch size in this context, with the results being shown in table 5.4 in the row called Run 3. This finding highlighted the sensitivity of DistilBERT to batch size, suggesting that smaller batch sizes may be more effective in achieving optimal performance, due to it benefiting more from the variability and frequent updates provided by smaller batch sizes.

In this phase of the research, the evaluation metrics used during training were aligned with those employed for testing the model, ensuring consistency across all stages of model assessment and evaluation. The compute_metrics function was defined to calculate several key performance metrics: precision, recall, F1-score, F2-score, and accuracy.

Subsequently, the Trainer class from the Hugging Face Transformers library was employed to manage the training and evaluation process. The Trainer was instantiated

with the DistilBERT model, the refined training arguments, the training and validation datasets, and the tokenizer. This configuration allowed for real-time computation of the specified metrics during both the training and evaluation phases, with this approach ensuring that the same set of metrics was consistently applied throughout the model's development, providing a robust framework for assessing improvements and fine-tuning the model.

By the end of the research into this model, the final set of training runs was executed, totaling eight runs, to solidify the results and validate the stability of the selected hyperparameters. Subsequently, efforts were made to utilize a fixed seed number on a dedicated server to ensure reproducibility and consistency in the training of DistilBERT. The average of these 8 runs is each with a different seed number and will be shown and discussed in the results subsection of DistilBERT (see Subsection 5.2.3).

This systematic approach to model training, characterized by iterative experimentation, hyperparameter optimization, and fine-tuning, underscores the importance of precise calibration in achieving optimal model performance. The findings from this research contribute valuable insights into the training dynamics of DistilBERT, particularly in the context of biomedical domain-specific tasks.

5.2.2 DistilBERT's testing

The testing script utilized to evaluate the DistilBERT model on a test set shares a significant resemblance to the one previously employed in the MonoBERT subsection (see Subsubsection 5.1.1.2). The script structure, processing steps, and overall methodology remain consistent, ensuring a standardized approach to model evaluation across different models. The MonoBERT testing script provided the foundation upon which the DistilBERT testing script was built. In the MonoBERT script, queries were run through the model to evaluate the relevance of documents, integrating query-specific information into the prediction process. However, in the DistilBERT testing script, this approach was modified. The predicted class for each document was now determined solely based on the knowledge acquired during training from relevant documents, independent of any particular query, rather than being influenced by documents relevant to the queries during the testing phase. This change was necessitated by practical considerations. Incorporating queries into the training process for DistilBERT would have required the model to evaluate each document in conjunction with every possible query, which would exponentially increase the computational complexity and time required for training. Such an approach would result in prohibitive training times, making it impractical for the scope of this dissertation.

Furthermore, optimization strategies were implemented in the DistilBERT script, particularly when dealing with document tokenization. Due to the model's 512 token limit, documents sporadically needed to be split into multiple chunks. However, unlike MonoBERT, where the score of a document was the average of the chunks' scores, DistilBERT adjustments led to significant scoring issues. If a document required splitting, the

score of the first chunk was predominantly used for final scoring since additional chunks often contained significantly fewer tokens (ranging from 5 to 20), producing markedly lower scores. This method arose because the low token count in subsequent chunks contributed negatively to the average score, thus penalizing the overall document score.

By omitting the query evaluation step and modifying the scoring approach, the DistilBERT model's testing script is designed to assess the model's ability to generalize from its training data to unseen documents, for a specific query, relying entirely on the learned representations of the text. This adjustment reflects a shift in focus from query-specific relevance to the broader task of document classification based on learned features. On the other hand, this focus allowed for the model to be trained for each biomarker dataset, instead of being a single biomedical domain model.

5.2.3 DistilBERT's results

The evaluation of the DistilBERT model's performance across several key metrics (see table 5.6) reveals mixed results, indicating some progress over previous models but still leaving room for improvement when compared to the previous results from the BLiR paper (see Table 2.3). The values display the average and standard deviation when applying these parameters (see Table 5.5, which contains all the parameters used to fine-tune this model). These results were obtained by running the model with seed numbers ranging from 1 to 10, which were chosen to ensure both reproducibility and an element of randomness in the evaluation. By averaging the results across these runs, the metrics presented in the figure represent a more reliable and generalized assessment of the model's performance, while the standard deviation provides insight into the variability and stability of these results.

	Precision	**Recall**	**F1-score**
DistilBERT's average	0.5505 ± 0.0237	0.6667 ± 0.0809	0.6002 ± 0.0317
Best BLiR results	**0.700**	**0.851**	**0.619**

	F2-score	**ROC AUC**	**NDCG@10**
DistilBERT's average	0.6375 ± 0.0574	**0.9566** ± 0.0062	0.7919 ± 0.1372
Best BLiR results	**0.689**	0.889	-

Table 5.6: Metrics performance of DistilBERT and Best BLiR results. The DistilBERT line represents an average of all runs with different seeds, while the Best BLiR line refers to different runs with different parameters that obtain the best value for each score. The BLiR paper did not report NDCG@10.

The Precision score of 0.5505 ± 0.0237 reflects an improvement over the MonoBERT model, yet it remains suboptimal compared to the values achieved in the BLiR paper as

evidenced by table 5.6. The observed value suggests that while the DistilBERT model has made strides in correctly identifying relevant documents, there is still a notable proportion of false positives, limiting its overall effectiveness.

Similarly, the Recall score, which is 0.6667 ± 0.0809, shows that the model can identify a higher proportion of true positives compared to MonoBERT. However, this metric also indicates underperformance, as it fails to meet the desired threshold established in the BLiR study. The relatively high standard deviation in retrieval metrics indicates considerable variability in performance across runs, suggesting that the model's reliability may fluctuate with different runs, thereby necessitating further tuning to enhance its stability and robustness.

The F1-score, a harmonic mean of Precision and Recall, stands at 0.6002 ± 0.0317. Although this represents an improvement over MonoBERT, it still is below the performance metrics reported in the BLiR paper subsection 2.4. Notably, the F1-score is close to the best achieved in the BLiR paper, falling short by about 1.9 percentage points. This proximity suggests that the model is beginning to achieve a similar performance in this metric, but further optimization would be needed to fully overcome the performance levels established in the BLiR study.

The F2-score, which weights recall higher than precision, has a value of 0.6375 ± 0.0574. This again indicates that while the model leans towards better recall, its overall performance is still lacking, particularly when more emphasis is placed on capturing as many relevant documents as possible. This outcome is critical in applications where missing relevant information could be costly.

Conversely, the ROC AUC score of 0.9566 ± 0.0062 is notably high, indicating that the DistilBERT model has a strong ability to distinguish between positive and negative classes across various threshold settings. This metric suggests that despite the underperformance in precision, recall, and F-scores, the model possesses robust discriminatory power, likely benefiting from the improvements over MonoBERT.

Among these runs, some runs results exceeded the one from BLiR. However, the fact that only these scores were maximized in some runs highlights the variability across different seeds, underscoring the need for further tuning and stability in the model's overall performance.

While the DistilBERT model shows some enhancements over MonoBERT, particularly in terms of discriminative capacity (as evidenced by the ROC AUC score), it still underperforms in key areas like precision, recall, and F-scores. These results indicate that, although some progress has been made, the model is not yet on par with the performance levels seen in the BLiR paper. This discrepancy created a necessity to explore an alternative model or approach to achieve the desired level of performance in document retrieval and classification tasks.

5.3 PubMedBERT

The final model chosen for this dissertation is PubMedBERT, Y.Gu et al [11], a model that stands out due to its pre-training on the PubMed corpus, being trained from scratch using 14 million PubMed abstracts, amounting to approximately 3 billion words. This corpus is directly relevant to the documents analyzed in this research, making PubMedBERT a strategic choice for tasks that demand a high level of biomedical understanding.

As the latest model in the sequence of those examined in this dissertation, PubMedBERT is based on BERT, a state-of-the-art transformer architecture that offers deep language understanding through an extensive pre-training process. This specialized focus on PubMed articles may represent the critical element that previous models lacked, as they were often pre-trained on general-domain text , potentially diluting their effectiveness in domain-specific tasks. By training exclusively on PubMed data, PubMedBERT avoids this dilution, resulting in a model finely tuned to the language and nuances of biomedical literature.

However, this advanced architecture also comes with a notable trade-off when compared to DistilBERT: the computational demands associated with BERT-based models are substantial. BERT's architecture is known for its large number of parameters and deep layers, which, while contributing to its powerful performance, also require significant computational resources for both training and inference.

This dissertation acknowledges the computational challenges posed by using this latest BERT-based model. Despite these challenges, the decision to use PubMedBERT is justified by its enhanced ability to process and understand the specialized language of biomedical literature. The computational resources required are seen as a necessary investment to achieve the highest possible accuracy and relevance in the NLP tasks central to this research, continuing the trend of selecting models that improve upon their predecessors.

By incorporating PubMedBERT into the research, the dissertation aims to leverage its robust BERT-based architecture, ensuring that the nuances of biomedical documents are well-understood and accurately processed. This approach continues the pattern of building upon earlier models, selecting the best available tools to meet the research objectives.

5.3.1 Fine-tuning and Optimization

The fine-tuning process for this research phase was conducted using the same training and testing scripts employed in the DistilBERT section (see Section 5.2), ensuring consistency in the experimental setup. No alterations were made to these scripts, allowing for a direct comparison of the results across different models. The model utilized in this phase is PubMedBERT, a highly specialized model pre-trained on both abstracts and full-text articles from PubMed.

5.3.1.1 Batch size

During fine-tuning, various batch sizes were tested to identify the optimal configuration for performance. It was observed that a batch size of 16 yielded the best results. When the batch size was increased to 32, the performance began to decline. This decline can be attributed to several factors inherent to fine-tuning large models like PubMedBERT.

One primary reason for this performance drop is related to the trade-off between batch size and the model's ability to generalize. Larger batch sizes often lead to faster convergence during training, which can sometimes result in the model converging to local minima that may not generalize well to unseen data. This phenomenon is particularly pronounced in complex models where overfitting can occur more easily with larger batch sizes.

Therefore, a batch size of 16 was selected as the optimal setting, balancing both the training efficiency and the model's ability to generalize effectively to new data.

5.3.1.2 Dropout Rate

Dropout is a regularization technique used during training to prevent overfitting, which occurs when a model performs well on training data but poorly on unseen data. By randomly dropping out a fraction of the units in a neural network layer during each forward and backward pass, dropout forces the model to learn more robust features that do not rely on specific neurons. The dropout rate is the proportion of units to drop, and it plays a crucial role in balancing the model's ability to generalize with its precision.

In this research, the dropout rate was experimented with at different levels—initially set at 0.1, then increased to 0.3, and finally to 0.5. The purpose of these adjustments was to identify the optimal dropout rate that would maximize the model's recall while maintaining high precision, thereby achieving the best possible F-scores.

The results indicated that increasing the dropout rate from 0.1 to 0.3 led to a noticeable improvement in recall. An increase in recall suggests that the model became better at capturing the broader range of relevant features from the input data. However, when the dropout rate was further increased to 0.5, although recall continued to improve, it came at the cost of precision. This trade-off resulted in a decline in overall model performance, as reflected in lower F-scores.

After careful evaluation, the dropout rate of 0.3 was deemed optimal. At this level, the model achieved a high recall without a substantial negative impact on precision. This balance allowed for better overall performance, as evidenced by the improved F-scores.

5.3.1.3 Epochs

In addition to tuning the dropout rate and batch size, the number of epochs was also adjusted during fine-tuning. The number of epochs determines how many times the learning algorithm will work through the entire training dataset. Initially, the model was

trained for 10 epochs as the DistilBERT model (see Section 5.2), followed by an increase to 20 epochs. While the results at 20 epochs showed improved performance, they were still not optimal.

The model was then trained for 50 epochs, which ultimately produced the best results. Training for more epochs allows the model to learn more from the data, but it also increases the risk of overfitting, where the model becomes too specialized to the training data and performs poorly on unseen data. Overfitting can occur because the model starts to adapt too effectively to the training data, capturing noise and outliers as if they were meaningful patterns.

However, in this conditions, even at 50 epochs, overfitting did not occur. This was likely due to several factors, including the use of dropout with an optimal rate of 0.3, which effectively prevented overfitting by ensuring that the model learned generalized patterns rather than relying on specific neurons. Furthermore, the underlying architecture of PubMedBERT, combined with domain-specific pre-training on the PubMed corpus, provided a solid foundation that helped the model maintain its performance without overfitting, even after extended training.

5.3.1.4 Cross-Validation

To further validate the model's robustness, a cross-validation approach with 5 folds was attempted. Cross-validation is a technique where the dataset is split into several parts named folds, with the model being trained on a combination of these folds while one fold is used for validation. This process is repeated several times which in this case was 5, with each fold serving as the validation set once, to ensure that the model's performance is consistent across different subsets of the data.

In this case, implementing 5-fold cross-validation significantly increased the training time, approximately fivefold, due to the need to train and validate the model multiple times. However, the results (see Table 5.7) did not vary considerably from those obtained with a single train-test split. There was a slight increase in recall, indicating a marginally better identification of relevant instances. However, similar to what was observed with other hyperparameters, this improvement in recall came at the expense of a slight reduction in precision. As a result, the overall F-scores did not show substantial improvement.

The findings suggest that while cross-validation can offer some benefits in terms of more robust model validation, in this particular case, the added computational cost did not yield sufficiently better results to justify its use over simpler validation methods. Therefore, while cross-validation was a useful exploratory tool, the final model configuration was determined without relying on it, favoring a balance between computational efficiency and performance.

	Precision	Recall	F1-score
Cross-validation Results	0.3099	**0.9362**	0.4656
PubMedBERT's average	**0.5364**	0.7979	**0.6378**

	F2-score	ROC AUC	NDCG@10
Cross-validation Results	0.6667	0.9651	-
PubMedBERT's average	**0.7237**	**0.9694**	0.8847

Table 5.7: Cross-validation run on the PubMedBERT model

5.3.1.5 Reduction of Non-relevants in the Dataset

During this research, several modifications were made to the training and testing datasets to explore their impact on model performance. The first experiment involved reducing the number of non-relevant articles in the training dataset. This reduction was performed in increments of 25%, 50%, and finally 75%. As the percentage of non-relevant articles decreased, the model's recall value showed a corresponding increase. This suggests that by reducing the noise from non-relevant articles, the model became more sensitive to identifying relevant instances within the dataset. However, as seen with previous adjustments, this improvement in recall came with a significant reduction in precision, similar to the effects observed with changes in other parameters.

	Precision	Recall	F1-score	F2-score	ROC AUC
Model's average	**0.5364**	0.7979	**0.6378**	0.7237	0.9694
25% Reduction	0.3613	**0.9149**	0.5181	0.7003	**0.9717**

	Precision	Recall	F1-score	F2-score	ROC AUC
50% Reduction	0.4505	0.8723	0.5942	**0.7348**	0.9692
75% Reduction	0.1621	1.0000	0.2789	0.4916	0.9693

Table 5.8: Reducing negatives run on the PubMedBERT model

Following this, a similar methodology was applied to the testing dataset. The non-relevant articles were reduced by the same percentages—25%, 50%, and 75%. Interestingly,

unlike with the training dataset, these reductions in the testing set resulted in no significant changes in the performance metrics. This outcome (see Table 5.8) suggests that while the training process may benefit from a cleaner dataset with fewer non-relevant instances, the testing phase, which assesses the model's generalization capabilities, remains largely unaffected by such modifications. This consistency in testing results underscores the robustness of the model's performance.

5.3.2 Results

The initial analysis involves the application of PubMedBERT to the Diet biomarkers dataset, serving as an experiment to establish comparable performance metrics to the one used in the BLiR paper (see Subsection 2.4 and Table 2.3).

Parameter	**Description**
`num_train_epochs`	Set to 50 epochs.
`per_device_train_batch_size`	Set to 16.
`per_device_eval_batch_size`	Set to 16.
`warmup_steps`	Set to 500 steps.
`weight_decay`	Set to 0.01.
`learning_rate`	Set to 1e-5.
`seed`	Run from 1 to 10.

Table 5.9: Training parameters used for optimizing BERT-Based models.

The selected training parameters (see Table 5.9) are designed to optimize the performance of the PubMedBERT model, particularly in the context of document retrieval within biomedical literature. These settings were chosen based on well-established findings from the literature, which provided a foundation for effective hyperparameter selection in similar tasks. Additionally, several experiments were conducted to fine-tune these parameters, ensuring they were useful to the model training. This combination of research-based and experimental approaches aimed to balance training efficiency with the need for comprehensive learning and robustness in performance metrics.

The chosen batch size of 16 for both training and evaluation strikes an effective balance between computational efficiency and model performance.This choice avoids the drawbacks of a too small a batch size, which would increase the computational load, and too large a batch size, which could diminish the model's ability to generalize well as well as perform better in unseen data. Changing to 32 reduces the performance of the model, as evidenced in Table 5.10 under line 32 Batch Size.

The implementation of 100 warmup steps (see Table 5.10 under line 100 Warmup Steps, which shows the results of lowering these warmup steps) was tested to examine its impact on the model's performance. While reducing the number of warmup steps allows the

learning rate to increase more quickly, the results indicate that it negatively affects the performance of the model. The rapid increase in learning rate from fewer warmup steps leads to larger updates early in the training process, potentially destabilizing the model and resulting in suboptimal performance.

In contrast, maintaining 500 warmup steps allows the learning rate to rise gradually, providing a more stable and controlled learning process. This approach prevents issues arising from abrupt large updates, which can harm model convergence and performance. Additionally, the application of a weight decay factor of 0.01 continues to aid in regularizing the model by penalizing larger weights, further improving generalizability and preventing overfitting.

A learning rate of 5e-6 (see Table 5.10 under line Learning rate changes, showcasing the results of lowering the learning rate to 5e-6) was applied to observe its effects on model fine-tuning. This more conservative learning rate slows down the optimization process, allowing for careful adjustments, particularly in highly specialized domains like biomedical research. While the use of 5e-6 does offer stable training, it also results in slower convergence and lower performance in comparison to a learning rate of 1e-5. A learning rate of 1e-5, while slightly more aggressive, provides a better balance between stability and speed. This rate allows for slow but effective fine-tuning without the excessive delay in convergence that comes with 5e-6. For models like PubMedBERT, which are applied to specialized domains, 1e-5 proves to be optimal, ensuring both accurate fine-tuning and faster convergence during training. Furthermore, to ensure reproducibility and assess the model's robustness, the training is conducted using seed values ranging from 1 to 10.

	Precision	**Recall**	**F1-score**	**F2-score**	**ROC AUC**
32 Batch Size	0.3667	**0.9362**	0.5269	0.7143	0.9665
100 Warmup Steps	0.3143	**0.9362**	0.4706	0.6707	**0.9711**
Learning Rate Change (5e-6)	**0.4019**	0.9149	**0.5584**	**0.7288**	0.9700

Table 5.10: PubMedBERT's performance metrics on various parameter changes

5.3.2.1 Results Structure

The subsection begins by presenting the results obtained using the PubMedBERT model alone, establishing a baseline for its performance in biomedical document retrieval tasks. The initial analysis focuses on evaluating PubMedBERT's capabilities without any dataset reduction techniques and on the Diet biomarkers, allowing for a clear understanding of the model's strengths and limitations in its purest form.

Following this, the influence of dataset reduction through BM25 integration on PubMedBERT's performance is scrutinized. This examination assesses whether a reduced dataset can maintain result integrity compared to those obtained from a full dataset,

thereby evaluating the trade-offs between computational efficiency and model effectiveness.

Finally, the analysis extends to PubMedBERT's performance on an expanded dataset that includes all the types of biomarkers. This setup evaluates the model's capacity to manage increased data volume and complexity, offering insights into its scalability and robustness, while also assessing its capability to assimilate the essential attributes of each biomarker and subsequently apply this knowledge effectively to the testing dataset.

5.3.2.2 PubMedBERT's results

	Precision	**Recall**	**F1-score**	**F2-score**	**ROC AUC**
Best BLiR results	**0.700**	**0.851**	0.619	0.689	0.889
MonoBERT's	0.0225 ± 0.0203	0.6944 ± 0.5833	0.0436 ± 0.0393	0.0996 ± 0.0893	0.6034 ± 0.4910
DistilBERT's	0.5505 ± 0.0237	0.6667 ± 0.0809	0.6002 ± 0.0317	0.6375 ± 0.0574	0.9566 ± 0.0062
PubMedBERT's	0.5364 ± 0.0445	0.7979 ± 0.0573	**0.6378** ± 0.0104	**0.7237** ± 0.0249	**0.9694** ± 0.0019
PubMedBERT + BM25	0.4836 ± 0.072	0.7842 ± 0.1128	0.587 ± 0.0401	0.6861 ± 0.0553	0.9629 ± 0.0036

Table 5.11: Performance metrics on the Diet Biomarker from every model used in this dissertation

The evaluation of the model's performance, as depicted in the table 5.11 in the line PubMedBERT's average, reveals several important insights into its efficacy in the context of document retrieval for biomedical literature, particularly in dietary biomarkers of exposure to environmental diseases. The model demonstrates a balanced performance across various metrics, with a particular emphasis on the F1 and F2 scores. These results are noteworthy, especially when compared to those reported in the BLiR paper (see Table 2.3 and line Best BLiR Results from Table 5.11).

In contrast to the findings in the BLiR paper, where recall tends to have higher values than precision, this model exhibits a more balanced trade-off between these two metrics. Although the recall and precision are slightly lower than some models, their closer alignment results in superior F1 and F2 scores. This balance is crucial as it indicates that the model is not overly skewed towards either high recall with low precision or vice versa, but rather maintains an equilibrium that enhances its overall effectiveness in retrieving relevant documents.

It is significant to note that while each performance metric from the BLiR paper—such as recall, precision, F1-score, F2-score, and ROC AUC- have had its highest value recorded

by different types of machine learning models in isolation, our model effectively concentrates many of these best results into a single framework. This consolidation of strong performance across multiple metrics into one model underscores its superiority over alternative approaches. It demonstrates that this model is not merely an average performer across different metrics, but rather excels by bringing together the best aspects typically found in separate and simpler models.

A notable observation is the precision metric, which, while being the lowest among the evaluated performance metrics and of the best value in the BLiR paper, remains comparable to other BERT-based models such as DistilBERT. Despite its relatively lower value, the precision achieved by this model is only marginally below that of similar architectures, indicating that the model does not sacrifice precision significantly to achieve its recall or vice-versa.

Moreover, all other performance metrics, including F1, F2, ROC AUC (0.9694 ± 0.0019), and NDCG@10 (0.8847 ± 0.069), outperform those of other BERT-based models tested on the same dataset. This suggests that the model not only excels in maintaining a balanced trade-off between recall and precision but also provides superior ranking and classification capabilities as reflected by the high ROC AUC and NDCG@10 scores.

The improvement in recall can be partially attributed to the adjustment of the dropout rate to 0.3, as explained in subsubsection 5.3.1.2. This value was previously identified as the optimal trade-off between precision and recall, and its application in this context has allowed the model to achieve a higher recall without a significant drop in precision. This change in dropout rate has proven to be a key factor in enhancing the model's recall and consequently F scores.

In conclusion, this model has achieved the best average performance across all BERT-based models tested in this dissertation on document retrieval within biomarker literature. The balanced and robust results, particularly in terms of F1 and F2 scores, highlight its effectiveness in accurately identifying relevant documents while maintaining a reasonable precision-recall trade-off.

5.3.2.3 BM25 + PubMedBERT

In this dissertation, the Okapi BM25 algorithm was utilized in conjunction with the PubMedBERT deep learning model to investigate the impact of dataset reduction on model performance. As discussed in a previous chapter, the BM25 algorithm is particularly effective in ranking documents based on their relevance, making it a suitable candidate for pre-processing steps aimed at reducing dataset size. Applying BM25, specifically the Okapi variant, which has been shown to outperform other versions in earlier evaluations (see Chapter 4), reduced the dataset by removing the bottom 20% of documents ranked by relevance. This approach was expected to decrease the computational resources required for training and testing without significantly compromising the model's performance.

The threshold chosen was, as mentioned earlier and shown in Chapter 4, the bottom 20% of documents ranked by relevance, using BM25 as the ranking function. This threshold was selected to retain all relevant documents while removing a significant portion of non-relevant ones. By applying a threshold to the bottom 20% of these BM25-ranked documents, the model can filter out lower-ranked, less relevant documents, which are likely to be non-relevant, while preserving the higher-ranked documents that are more likely to contain relevant content, reducing the number of documents available to train the model. This approach balances the need to reduce non-relevant instances without sacrificing the retrieval of important, relevant documents, thus optimizing the model's performance.

The results show that PubMedBERT + BM25 (presented in Table 5.11 in the line called PubMedBERT + BM25) performs slightly worse than PubMedBERT alone across most metrics, with a noticeable increase in standard deviation. Precision, Recall, F1-score, and F2-score are all lower for PubMedBERT + BM25, indicating a reduced ability to identify relevant documents and maintain a balance between precision and recall. Additionally, the higher standard deviations across these metrics suggest that the inclusion of BM25 introduces more variability and less consistent performance. Although both models perform well in ROC AUC and NDCG@10, PubMedBERT without BM25 demonstrates slightly better performance and greater consistency.

Despite the computational advantages gained through dataset reduction using BM25 as the reduced number of documents allowed for faster training, the overall performance did not improve as expected when paired with the PubMedBERT model. The results underscore a critical observation: BERT-based models, such as PubMedBERT, generally perform better with larger datasets, allowing for more comprehensive learning during the training phase. The reduction of the dataset, although beneficial for managing computational resources, appears to hinder the model's ability to fully leverage its learning capacity, leading to suboptimal performance during testing.

Furthermore, as detailed in earlier sections (see Subsubsection 5.3.1.5), the strategy of removing non-relevant documents increases recall at the cost of precision, which is reflected in the higher F2-score and lower F1-score. While this may be advantageous in scenarios where recall is prioritized, it ultimately diminishes the balance between precision and recall, which is crucial for achieving high F1-scores.

Given these findings, the use of BM25 for dataset reduction in the context of PubMedBERT appears to be redundant. While BM25 can effectively reduce the dataset size and computational load, it does not translate into better model performance. Instead, it may be more advantageous to utilize the full dataset, allowing PubMedBERT to maximize its learning potential and achieve superior results in both the training and testing phases.

Another solution, the DistilBERT model emerges as a more effective alternative. This streamlined version of BERT not only achieves higher precision and F1 scores compared

to PubMedBERT + BM25, but also maintains a closer balance between precision and recall. These outcomes suggest that DistilBERT is a more moderate and effective solution for document retrieval compared to BM25, which tends to exhibit a high recall but at the expense of precision. Thus, DistilBERT's performance characteristics indicate that it can maintain efficacy even with reduced datasets due to its distillation techniques, effectively balancing computational efficiency with robust model performance.

This analysis underscores the need to carefully consider the balance between dataset size and model capabilities. Utilizing the full capabilities of advanced BERT-based models like PubMedBERT may necessitate larger datasets to truly capitalize on their sophisticated learning mechanisms, rather than compromising with dataset reduction techniques that do not yield the expected performance enhancements.

5.3.2.4 All biomarkers's results

	Precision	**Recall**	**F1-score**	**F2-score**	**ROC AUC**
PubMedBERT's	**0.4105** ± 0.035	**0.6403** ± 0.07	**0.4962** ± 0.0143	**0.5717** ± 0.0314	**0.9433** ± 0.0022
BLiR All biomarkers	0.354	0.630	0.453	0.545	0.781

Table 5.12: PubMedBERT's performance on all the Biomarkers

The table above (see Table 5.12) presents the average score as well as the standard deviation of the performance metrics used in this dissertation, including Precision, Recall, F1-score, F2-score, ROC AUC, and NDCG@10.

The model shows moderate performance in identifying relevant instances, with a Precision of 0.4105 and a consistent standard deviation of ± 0.035, indicating stable precision across scenarios. Recall, averaging 0.6403 with a higher standard deviation of ± 0.07, reflects the model's effectiveness in capturing relevant instances, though with more variability. The F1-score of 0.4962 and its lower standard deviation of ± 0.0143 suggest a more stable balance between Precision and Recall. Similarly, the F2-score emphasizes Recall, averaging 0.5717 with a standard deviation of ± 0.0314. The ROC AUC demonstrates excellent discriminative power with a high average of 0.9433 and minimal variability (± 0.0022). However, the NDCG@10, with an average of 0.7798 and a higher standard deviation of ± 0.0796, indicates that the model's ranking quality is more variable, depending on the dataset or biomarker.

Given that the point of comparison for this model is the one used in the BLiR paper, which focuses on diet biomarkers, it is important to acknowledge that documents relevant to each biomarker can vary significantly from one another. This inherent variability among biomarkers is reflected in the performance metrics, particularly in the Recall and

NDCG@10, where the standard deviations are higher. The model tends to perform better when only one biomarker is present, likely due to the reduced complexity and variability in the data. Therefore, the results presented here are to be expected given the nature of the input data.

Additionally, this dissertation's results on all biomarkers were worse than those observed when focusing solely on diet biomarkers, which is consistent with the findings from the original BLiR paper. This reinforces the idea that the model struggles with greater variability across biomarkers and performs better when tailored to a single biomarker. Variations in what constitutes relevant documents for each biomarker further complicate the task. The intricate nature of biomarkers means that what is considered relevant for one biomarker might differ substantially from what is relevant for another. This variability can explain the lower performance scores, even when using the best-performing model. The model's difficulty in consistently identifying and ranking the most relevant files across different biomarkers is a reflection of these underlying complexities.

A potential solution to improve the model's performance across multiple biomarkers would be to apply the model separately to each biomarker type, rather than using a single model for all biomarkers collectively. This approach could help mitigate the variability by tailoring the model's analysis to the specific characteristics of each biomarker type, thereby improving the overall accuracy and consistency of the results. However, the Exposome-Explorer database is still limited in terms of other biomarker types when compared to the Diet biomarker.

5.4 Final Thoughts

This chapter has provided an extensive analysis of BERT-based models within the context of document retrieval for biomedical literature, focusing on three distinct models: MonoBERT, DistilBERT, and PubMedBERT. Each model represents a different adaptation of the original BERT framework, tailored to meet specific requirements and address various challenges associated with biomedical text processing and document retrieval.

MonoBERT was initially explored due to its fine-tuning for document retrieval tasks, specifically within the biomedical domain. However, its application in this dissertation was limited by several factors, including the availability of only its large version, which imposed significant computational demands. The testing process revealed that while MonoBERT had potential, its resource-intensive nature made it less practical for extensive use in this research without access to high-performance computing resources like CUDA-enabled GPUs. Despite these limitations, the insights gained from MonoBERT's implementation were valuable, particularly in understanding the importance of computational efficiency in deploying deep learning models for large-scale document retrieval.

Following the challenges encountered with MonoBERT, DistilBERT was selected as a more computationally efficient alternative. DistilBERT, a distilled version of BERT, provided a balance between performance and resource efficiency, making it a more

feasible option given the available computational resources. The fine-tuning of DistilBERT allowed for more extensive experimentation with hyperparameters, leading to notable improvements in performance metrics such as F1-score, F2-score, and ROC AUC. However, despite these improvements, DistilBERT still exhibited variability in its results, particularly in recall and precision, indicating that while it was a significant step forward, further optimization was needed.

The final model examined in this chapter, PubMedBERT, was specifically pre-trained on PubMed corpora, making it particularly well-suited to the domain-specific tasks addressed in this dissertation. PubMedBERT demonstrated superior performance in capturing the nuances of biomedical text, largely due to its specialized pre-training on the PubMed corpus. Fine-tuning PubMedBERT involved careful adjustments to batch size, dropout rate, and the number of training epochs, which were optimized to balance training efficiency with the model's ability to generalize effectively. The results indicated that PubMedBERT outperformed the other models, particularly in terms of its F1 and F2 scores, while maintaining high recall and average precision.

Furthermore, PubMedBERT's integration with the BM25 algorithm, though effective in reducing computational complexity, revealed some limitations. While BM25 contributed to dataset reduction, it did not significantly enhance model performance, as PubMedBERT's ability to leverage its specialized pre-training appeared more effective on larger datasets. This result highlights a trade-off between computational efficiency and the need for extensive training data, underscoring PubMedBERT's particular strength in handling comprehensive biomedical datasets. When applied to an expanded dataset containing all biomarkers, PubMedBERT's performance remained robust, though it was somewhat challenged by the variability among different biomarkers. This observation aligns with previous work on BLiR, where results on all biomarkers also fell short of those focused solely on diet biomarkers, further validating the impact of domain-specific focus in model performance.

In conclusion, the exploration of these BERT-based models has highlighted the importance of selecting a model that aligns closely with the specific demands of the task at hand. This chapter's findings underscore the critical role of model selection and fine-tuning in achieving optimal performance in complex NLP tasks, particularly within specialized domains like biomedical literature.

6

Final Remarks

This dissertation builds upon and progresses the work previously done in the BLiR framework by further enhancing the document retrieval process for the Exposome-Explorer database, with the primary aim of reducing the time and effort required by researchers in the biomedical field to retrieve relevant literature. Through the careful selection and integration of standard and transformer-based models, this research has successfully addressed key challenges identified in the BLiR paper and improved upon its results.

BM25 was initially explored for filtering out less relevant documents and streamlining the dataset for subsequent analysis. However, its use proved less effective than anticipated, as the transformer model utilized in this thesis, PubMedBERT, was negatively impacted by the dataset reduction. The loss of data through BM25 filtering hindered the models' ability to leverage their full capacity for understanding context-rich biomedical text. Consequently, BM25 was ultimately excluded from the final pipeline to focus on optimizing the transformer models for improved efficiency and performance without the need for dataset reduction.

The analysis of transformer-based models further contributed to this optimization. While MonoBERT demonstrated potential for document retrieval, its resource-intensive nature limited its practical application within the computational constraints of this research. DistilBERT, on the other hand, offered a more balanced approach between performance and computational efficiency, yet still required further optimization to address its variability in recall and precision.

PubMedBERT emerged as the most effective model, particularly due to its pre-training on domain-specific PubMed corpora. Its ability to generalize across a wide variety of biomedical texts, coupled with careful fine-tuning, resulted in the highest overall performance, making it the most suitable model for this dissertation's objectives. Despite some challenges with the integration of BM25 in reducing computational complexity, PubMedBERT's specialized capabilities in handling large datasets and capturing domain-specific nuances affirmed its superior performance in biomedical literature retrieval.

For future work, this dissertation proposes applying the models developed and fine-tuned during the research to the most recent version of the Exposome-Explorer database.

This would involve collaboration with the researchers managing the database to retrieve all the necessary queries and documents, ensuring that the models are tested and validated in a real-world, updated setting. Additionally, a promising avenue for improving the performance of the BM25 model lies in the integration of RM3 (as highlighted in Subsubsection 2.7.1.2). RM3, a relevance feedback method, could enhance BM25 by expanding the list of terms or queries associated with each document, enriching the term space, and allowing the model to capture more relevant contextual information. This would likely improve document ranking and retrieval effectiveness, leading to more comprehensive and accurate results in biomedical literature searches.

Moreover, further optimization of the deep learning models used in this research is planned. Fine-tuning hyperparameters, exploring new training strategies, and leveraging more recent model architectures could enhance model performance. Additionally, the developed pipeline will also be tested on other biomedical database curation scenarios.

In conclusion, this dissertation has demonstrated that the use of state-of-the-art deep learning models can significantly improve the efficiency and accuracy of document retrieval in the biomedical domain. The work highlights the importance of selecting and fine-tuning models according to task-specific demands, ultimately contributing to more effective solutions for database curation in biomarker research.

Bibliography

[1] D. R. Cheriton. "From doc2query to docTTTTTquery". In: 2019. URL: https://api.semanticscholar.org/CorpusID:208612557 (cit. on p. 18).

[2] F. Couto and A. Lamurias. "MER: A shell script and annotation server for minimal named entity recognition and linking". In: *Journal of Cheminformatics* 10 (2018-12). DOI: 10.1186/s13321-018-0312-9 (cit. on p. 16).

[3] D. Brown. *rank-bm25*. https://pypi.org/project/rank-bm25/. Accessed: 2024-08-06. 2022 (cit. on p. 41).

[4] J. Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL] (cit. on pp. 1, 11–13, 27, 52).

[5] R. Dogan, R. Leaman, and Z. lu. "NCBI Disease Corpus: A Resource for Disease Name Recognition and Concept Normalization". In: *Journal of biomedical informatics* 47 (2014-01). DOI: 10.1016/j.jbi.2013.12.006 (cit. on p. 14).

[6] *Exposome-Explorer*. http://exposome-explorer.iarc.fr/. Accessed on: 24/01/2024 (cit. on p. 14).

[7] T. Formal et al. "SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval". In: *ArXiv* abs/2109.10086 (2021). URL: https://api.semanticscholar.org/CorpusID:237581550 (cit. on pp. 19–21, 27).

[8] L. Gao and J. Callan. "Condenser: a Pre-training Architecture for Dense Retrieval". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Ed. by M.-F. Moens et al. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, 2021-11, pp. 981–993. DOI: 10.18653/v1/2021.emnlp-main.75. URL: https://aclanthology.org/2021.emnlp-main.75 (cit. on p. 21).

[9] L. Gao and J. Callan. "Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by S. Muresan, P. Nakov, and A. Villavicencio. Dublin, Ireland: Association for Computational

Linguistics, 2022-05, pp. 2843–2853. DOI: 10.18653/v1/2022.acl-long.203. URL: https://aclanthology.org/2022.acl-long.203 (cit. on pp. 21, 22, 27).

[10] M. Gospodinov, S. MacAvaney, and C. Macdonald. "Doc2Query–: When Less is More". In: *Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, 2023, Proceedings, Part II*. Dublin, Ireland: Springer-Verlag, 2023, pp. 414–422. ISBN: 978-3-031-28237-9. DOI: 10.1007/978-3-031-28238-6_31. URL: https://doi.org/10.1007/978-3-031-28238-6_31 (cit. on pp. 18, 19, 27).

[11] Y. Gu et al. "Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing". In: *ACM Transactions on Computing for Healthcare* 3.1 (2021-10), pp. 1–23. ISSN: 2637-8051. DOI: 10.1145/3458754. URL: http://dx.doi.org/10.1145/3458754 (cit. on pp. 4, 25, 52, 53, 64).

[12] N. A. Jaleel et al. "UMass at TREC 2004: Novelty and HARD". In: *Text Retrieval Conference*. 2004. URL: https://api.semanticscholar.org/CorpusID:16221853 (cit. on p. 31).

[13] K. KAGEURA and B. Umino. "Methods of Automatic Term Recognition — A Review —". In: *Terminology* 3 (2001-02). DOI: 10.1075/term.3.2.03kag (cit. on p. 31).

[14] A. Lamurias et al. "Information Retrieval Using Machine Learning for Biomarker Curation in the Exposome-Explorer". In: *Frontiers in Research Metrics and Analytics* 6 (2021). ISSN: 2504-0537. DOI: 10.3389/frma.2021.689264. URL: https://www.frontiersin.org/articles/10.3389/frma.2021.689264 (cit. on pp. 2, 15, 17, 56).

[15] LasigeBioTM. *BLiR*. https://github.com/lasigeBioTM/BLiR/tree/master. Accessed: 2024-08-06. 2024 (cit. on p. 39).

[16] V. Lavrenko and W. B. Croft. "Relevance-Based Language Models". In: *ACM SIGIR Forum* 51 (2001), pp. 260–267. URL: https://api.semanticscholar.org/CorpusID:14116318 (cit. on p. 31).

[17] J. Lee et al. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining". In: *Bioinformatics* 36.4 (2019-09), pp. 1234–1240. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btz682. eprint: https://academic.oup.com/bioinformatics/article-pdf/36/4/1234/48983216/bioinformatics_36_4_1234.pdf. URL: https://doi.org/10.1093/bioinformatics/btz682 (cit. on p. 13).

[18] S. Lim and J. Kang. "Chemical–gene relation extraction using recursive neural network". In: *Database* 2018 (2018-06), bay060. ISSN: 1758-0463. DOI: 10.1093/database/bay060. eprint: https://academic.oup.com/database/article-pdf/doi/10.1093/database/bay060/27438554/bay060.pdf. URL: https://doi.org/10.1093/database/bay060 (cit. on p. 14).

[19] S.-C. Lin, J.-H. Yang, and J. Lin. "In-Batch Negatives for Knowledge Distillation with Tightly-Coupled Teachers for Dense Retrieval". In: *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*. Ed. by A. Rogers et al. Online: Association for Computational Linguistics, 2021-08, pp. 163–173. DOI: 10.18653/v1/2021.repl4nlp-1.17. URL: https://aclanthology.org/2021.repl4nlp-1.17 (cit. on p. 18).

[20] Y. Lv and C. Zhai. "When documents are very long, BM25 fails!" In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM. 2011, pp. 1103–1104 (cit. on pp. 42, 44).

[21] V. Nguyen, M. Rybinski, and S. Karimi. "Matching a Patient from An Admission Note to Clinical Trials: Experiments with Query Generation and Neural-Ranking". In: *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022*. Ed. by I. Soboroff and A. Ellis. Vol. 500-338. NIST Special Publication. National Institute of Standards and Technology (NIST), 2022. URL: https://trec.nist.gov/pubs/trec31/papers/CSIROmed.T.pdf (cit. on pp. 31, 32, 34).

[22] R. Nogueira and K. Cho. "Passage Re-ranking with BERT". In: *ArXiv* abs/1901.04085 (2019). URL: https://api.semanticscholar.org/CorpusID:58004692 (cit. on pp. 4, 22, 52, 53, 56).

[23] G. M. di Nunzio, G. Faggioli, and S. Marchesin. "Summarize and Expand Queries in Clinical Trials Retrieval. The IIIA Unipd at TREC 2022 Clinical Trials". In: *Text Retrieval Conference*. 2022. URL: https://api.semanticscholar.org/CorpusID:261288483 (cit. on p. 31).

[24] M. E. Peters et al. "Deep Contextualized Word Representations". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Ed. by M. Walker, H. Ji, and A. Stent. New Orleans, Louisiana: Association for Computational Linguistics, 2018-06, pp. 2227–2237. DOI: 10.18653/v1/N18-1202. URL: https://aclanthology.org/N18-1202 (cit. on p. 11).

[25] R. Pradeep, R. Nogueira, and J. J. Lin. "The Expando-Mono-Duo Design Pattern for Text Ranking with Pretrained Sequence-to-Sequence Models". In: *ArXiv* abs/2101.05667 (2021). URL: https://api.semanticscholar.org/CorpusID:231603106 (cit. on p. 18).

[26] R. Pradeep et al. "Neural Query Synthesis and Domain-Specific Ranking Templates for Multi-Stage Clinical Trial Matching". In: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '22. <conf-loc>, <city>Madrid</city>, <country>Spain</country>, </conf-loc>: Association for Computing Machinery, 2022, pp. 2325–2330. ISBN: 9781450387323. DOI: 10.1145/3477495.3531853. URL: https://doi.org/10.1145/3477495.3531853 (cit. on p. 33).

[27] R. Pradeep et al. "Squeezing Water from a Stone: A Bag of Tricks for Further Improving Cross-Encoder Effectiveness for Reranking". In: *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I*. Stavanger, Norway: Springer-Verlag, 2022, pp. 655–670. ISBN: 978-3-030-99735-9. DOI: 10.1007/978-3-030-99736-6_44. URL: https://doi.org/10.1007/978-3-030-99736-6_44 (cit. on p. 18).

[28] Y. Qu et al. "RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by K. Toutanova et al. Online: Association for Computational Linguistics, 2021-06, pp. 5835–5847. DOI: 10.18653/v1/2021.naacl-main.466. URL: https://aclanthology.org/2021.naacl-main.466 (cit. on p. 21).

[29] A. Radford et al. *Improving Language Understanding with Unsupervised Learning*. Technical Report. OpenAI, 2018 (cit. on p. 11).

[30] C. Raffel et al. *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. 2023. arXiv: 1910.10683 [cs.LG] (cit. on p. 18).

[31] K. Roberts et al. "Overview of the TREC 2022 Clinical Trials Track". In: *Proceedings of the Thirty-First Text REtrieval Conference, TREC 2022, online, November 15-19, 2022*. Ed. by I. Soboroff and A. Ellis. Vol. 500-338. NIST Special Publication. National Institute of Standards and Technology (NIST), 2022. URL: https://trec.nist.gov/pubs/trec31/papers/Overview%5C_trials.pdf (cit. on pp. 3, 28–30, 53).

[32] S. Robertson and H. Zaragoza. "The Probabilistic Relevance Framework: BM25 and Beyond". In: 3.4 (2009-04), pp. 333–389. ISSN: 1554-0669. DOI: 10.1561/1500000019. URL: https://doi.org/10.1561/1500000019 (cit. on pp. 1, 6, 27, 42).

[33] S. E. Robertson and H. Zaragoza. "The BM25 formula and its extensions". In: *Proceedings of the ACM SIGIR Forum*. Vol. 48. 1. ACM. 2014, pp. 59–66. URL: https://www.cs.otago.ac.nz/homepages/andrew/papers/2014-2.pdf (cit. on p. 41).

[34] M. Rybinski, J. Xu, and S. Karimi. "Clinical trial search: Using biomedical language understanding models for re-ranking". In: *Journal of Biomedical Informatics* 109 (2020), p. 103530. ISSN: 1532-0464. DOI: https://doi.org/10.1016/j.jbi.2020.103530. URL: https://www.sciencedirect.com/science/article/pii/S1532046420301581 (cit. on p. 32).

[35] V. Sanh et al. *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. 2020. arXiv: 1910.01108 [cs.CL] (cit. on pp. 4, 23, 25, 52).

[36] G. Tsatsaronis et al. "An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition". In: *BMC Bioinformatics* 16 (2015-04), p. 138. DOI: 10.1186/s12859-015-0564-6 (cit. on p. 14).

[37] Ö. Uzuner et al. "2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text". In: *Journal of the American Medical Informatics Association : JAMIA* 18 5 (2011), pp. 552–6. URL: https://api.semanticscholar.org/CorpusID:30029552 (cit. on p. 14).

[38] A. Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (cit. on pp. 10, 17).

[39] X. Wang et al. "Cross-type biomedical named entity recognition with deep multi-task learning". In: *Bioinformatics* 35.10 (2018-10), pp. 1745–1752. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/bty869. eprint: https://academic.oup.com/bioinformatics/article-pdf/35/10/1745/48970159/bioinformatics_35_10_1745.pdf. URL: https://doi.org/10.1093/bioinformatics/bty869 (cit. on p. 14).

[40] *Web of Science*. Accessed on: 24/01/2024 (cit. on p. 15).

[41] Y. Wu et al. *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. 2016. arXiv: 1609.08144 [cs.CL] (cit. on p. 12).

[42] G. Xu et al. "Hybrid Retrieval and Multi-stage Text Ranking Solution at TREC 2022 Deep Learning Track". In: *Proceedings of the 31st Text REtrieval Conference (TREC)*. Accessed: 2024-09-23. National Institute of Standards and Technology (NIST). 2022. URL: https://trec.nist.gov/pubs/trec31/papers/Ali.D.pdf (cit. on p. 27).

I

Biomarkers Queries

I.1 Diet Biomarker Queries

evaluation, validity, validation, valid, reliability, recall, intake, consumption, serum, blood, plasma, urine, urinary, adipose-tissue, hair, intake, consumption, diet, dietary, biomarker, marker, indice, indicators, intake, consumption, diet, dietary, comparison, compared, association, associated, correlation, correlated, relation, metabolites, concentration, excretion, levels, serum, blood, plasma, urine, urinary, adipose-tissue, hair, evaluation, validity, validation, valid, reliability, questionnaire, recall, consumption, serum, blood, plasma, urine, urinary, adipose-tissue, hair, nutrients, fatty-acid, pufa, lipids, hca, vitamins, carotenoids, sugars, glucosinolates, alcohols, enterolactones, phytoestrogens, isoflavones, isothiocyanates, alkylresorcinols, resorcinols, cholesterols, vitisin, resveratrol, stilbenes, polyphenols, phenols, flavon, flavan, cinnamic, benzoic, anthocyan, quercetin, luteolin, myricetin, apigenin, isorhamnetin, catechins, epicatechins, epigallocatechins, naringin, hesperidin, lignans, tannins, ellagitannins, ellagic, kaempferol, proanthocyanidins, procyanidins, caffeic, ferulic, sinapic, chlorogenic, salt, potassium, calcium, nitrogen, copper, magnesium, selenium, zinc, phospholipids, cryptoxanthin, carotene, lycopene, lutein, zeaxanthin, tocopherol, retinol, ascorbic, folic, homocysteine, fibers, linolenic, linoleic, saponins, methylxanthins, phytosterols, polyamines, proteins, amino-acids, betalains, terpenes, sulfurs, allicin, organicacids, indoles, benzopyrroles, eicosapentaenoic, docosahexaenoic, folate, thiamine, riboflavin, niacin, iron, iodine, sodium, ferritin, lipoproteins, triacylglycerols, oleic, palmitic, stearic, phosphatidyl, sphingomyelin, chalcones, dihydrochalcones, isoflavonoids, flavonoids, coumarins, diet, dietary, foods, citrus, oranges, pomelos, grapefruits, lemons, tang, apples, apricots, cherries, quinces, peaches, nectarines, pears, plums, prunes, berries, currants, gooseberries, bilberries, blackberries, blueberries, cranberries, raspberries, strawberries, redcurrants, whitecurrants, blackcurrants, greencurrants, vinegars, grapes, wines, vinifera, raisins, grapeseed-oil, pomegranates, persimmons, bananas, mangos, pineapples, kiwis, figs, guavas, star-apples, star-fruits, starapples, star-fruits, papayas, avocados, melons, litchis, exotic-fruits, tropical-fruits, passion-fruits,

fruits, vegetables, fishes, meats, cereals, dairies, beverages, grains, nuts, brassicas, cabbages, herbs, spices, coffees, eggs, oils, gourds, milks, pulses, peas, beans, lentils, teas, onions, garlics, pomes, drupes, cocoas, chocolates, beef, pigs, swines, poultry, chickens, ducks, goats, cheeses, mushrooms, candies, seasonings, breads, condiments, cuisines, pastas, potatoes, wheats, ryes, rices, oats, barleys, buckwheats, beers, olives, maizes, corns, artichokes, carrots, celeries, chicories, lettuces, tomatoes, seafoods, mints, apiaceae, parsleys, soy, soyas, soyabeans, soybeans, tofu, intakes, consumptions, biomarkers, markers, indices, indicators, evaluations, validity, validation, reliability, FFQ, questionnaires, recalls, biomarkers, markers, indicators, consumptions, and, serum, blood, plasma, urine, urinary, hair, nails, toenails, erythrocytes, RBC, adipose-tissue, skin, breath, saliva, faeces, feces, and, correlated, correlations, associations, associated, relationships, concentrations.

I.2 Pollutant Biomarkers Queries

I.2.1 DPBS Biomarker

disinfection byproduct, disinfection by-product, chloromethane, bromomethane, halomethane, haloacetic acid, chloroacetic acid, bromoacetic acid, haloaldehyde, bromoacetaldehyde, chloroacetaldehyde, haloketone, chloroacetone, chloropropanone, haloacetonitrile, bromoacetonitrile, chloroacetonitrile, bromoform, chloroform, urine, urinary, serum, plasma, blood, adduct, biological monitoring level, excretion, exposure, exposed, biomarker, biomarkers, marker, markers, metabolite, metabolites, concentration, concentrations.

I.2.2 HCA Biomarker

hca, heterocyclic, amine, heterocyclic, amines, heterocyclic, aromatic, amine, heterocyclic, aromatic, amines, phip, urine, urinary, serum, plasma, blood, adduct, biological, monitoring, level, excretion, exposure, exposed, biomarker, biomarkers, marker, markers, metabolite, metabolites, concentration, concentrations, urine, urinary, serum, plasma, blood

I.2.3 PAH Biomarker

pah, polycyclic, aromatic, hydrocarbons, hydroxypyrene, urine, urinary, serum, plasma, blood, adduct, biological, monitoring, level, excretion, exposure, exposed, biomarker, biomarkers, marker, markers, metabolite, metabolites, concentration, concentrations, urine, urinary, serum, plasma, blood, adduct

I.2.4 PCB Biomarker

pcb, pcbs, polychlorinated, biphenyls, urine, urinary, serum, plasma, blood, adduct, biological, monitoring, level, excretion, exposure, exposed, biomarker, biomarkers, marker,

markers, metabolite, metabolites, concentration, concentrations, urine, urinary, serum, plasma, blood, adduct

I.2.5 Phtalates Biomarker

phthalate, urine, urinary, serum, plasma, blood, adduct, biological, monitoring, level, excretion, exposure, exposed, biomarker, biomarkers, marker, markers, metabolite, metabolites, concentration, concentrations, urine, urinary, serum, plasma, blood, adduct

I.2.6 Polybrominated Biomarker

polybrominated, diphenyl, ethers, polybrominated, diphenylethers, pbde, pbdes, polybrominated, biphenyl, pbb, pbbs, urine, urinary, serum, plasma, blood, adduct, biological, monitoring, level, excretion, exposure, exposed, biomarker, biomarkers, marker, markers, metabolite, metabolites, concentration, concentrations, urine, urinary, serum, plasma, blood, adduct

I.2.7 Polychlorinated Biomarker

pcdd, pcdf, polychlorinated dibenzo-p-dioxins, dibenzofurans, urine, urinary, serum, plasma, blood, adduct, biological monitoring level, excretion, exposure, exposed, biomarker, biomarkers, marker, markers, metabolite, metabolites, concentration, concentrations.

I.3 Reproducibility Biomarker Keywords

excretion, levels, serum, blood, plasma, urine, urinary, adipose-tissue, hair, blood pressure, reliability, reproducibility, repeatability, variability, intra-individual, inter-individual, intraindividual, interindividual, within-individual, between-individual, intra-subject, inter-subject, within-subject, between-subject, within-person, between-person, serum, blood, plasma, urine, urinary, adipose-tissue, hair, excretion, levels, serum, blood, plasma, urine, urinary, adipose-tissue, hair, blood pressure, reliability, reproducibility, repeatability, variability, intra-individual, inter-individual, intraindividual, interindividual, within-individual, between-individual, intra-subject, inter-subject, within-subject, between-subject, within-person, between-person, serum, blood, plasma, urine, urinary, adipose-tissue, hair, nutrients, fatty-acid, pufa, lipids, hca, vitamins, carotenoids, sugars, glucosinolates, alcohols, enterolactones, phytoestrogens, isoflavones, isothiocyanates, alkylresorcinols, resorcinols, cholesterols, vitisin, resveratrol, stilbenes, polyphenols, phenols, flavon, flavan, cinnamic, benzoic, anthocyan, quercetin, luteolin, myricetin, apigenin, isorhamnetin, catechins, epicatechins, epigallocatechins, nar, hes, lignans, tannins, ellagitannins, ellagic, kaempferol, proanthocyan, procyan, caffeic, ferulic, sinapic, chlorogenic, salt, potassium, calcium, nitrogen, copper, magnesium, selenium, zinc, phospholipids, cryptoxanthin, carotene, lycopene, lutein, zeaxanthin, tocopherol, retinol, ascorbic,

folic, homocysteine, fibers, linolenic, linoleic, saponins, methylxanthins, phytosterols, polyamines, proteins, amino-acids, betalains, terpenes, sulfurs, allicin, organicacid, indoles, benzopyrroles, eicosapentaenoic, docosahexaenoic, folate, thiamine, riboflavin, niacin, iron, iodine, sodium, ferritin, lipoproteins, triacylglycerols, oleic, palmitic, stearic, phosphatidyl, sphingomyelin, chalcones, dihydrochalcones, isoflavonoids, flavonoids, coumarins, biomarkers, markers, indices, indicators, metabolites, concentrations, exposure, exposed, serum, blood, plasma, urine, urinary, hair, nails, toenails, erythrocytes, RBC, adipose-tissue, skin, breath, saliva, faeces, feces, reliability, reproducibility, variability, nutrient, nutrients, fatty-acid, pufa, lipid, lipids, hca, vitamin, vitamins, carotenoid, carotenoids, sugar, sugars, glucosinolate, glucosinolates, alcohol, alcohols, enterolactone, enterolactones, phytoestrogen, phytoestrogens, isoflavone, isoflavones, isothiocyanate, isothiocyanates, alkylresorcinol, alkylresorcinols, resorcinol, resorcinols, cholesterol, cholesterols, vitisin, resveratrol, stilbene, polyphenol, phenol, flavon, flavan, cinnamic, benzoic, anthocyan, quercetin, luteolin, myricetin, apigenin, isorhamnetin, catechin, epicatechin, epigallocatechin, nar, hes, lignan, tannin, ellagitannin, ellagic, kaempferol, proanthocyan, procyan, caffeic, ferulic, sinapic, chlorogenic, salt, potassium, calcium, nitrogen, copper, magnesium, selenium, zinc, phospholipid, phospholipids, cryptoxanthin, carotene, lycopene, lutein, zeaxanthin, tocopherol, retinol, ascorbic, folic, homocysteine, fiber, fibers, linolenic, linoleic, saponin, saponins, methylxanthin, methylxanthins, phytosterol, phytosterols, polyamine, polyamines, protein, proteins, amino-acid, betalain, betalains, terpene, terpenes, sulfur, allicin, organicacid, indole, indoles, benzopyrrole, benzopyrroles, eicosapentaenoic, docosahexaenoic, folate, thiamine, riboflavin, niacin, iron, iodine, sodium, ferritin, lipoprotein, lipoproteins, triacylglycerol, triacylglycerols, oleic, palmitic, stearic, phosphatidyl, sphingomyelin, chalcone, chalcones, dihydrochalcone, dihydrochalcones, isoflavonoid, isoflavonoids, flavonoid, flavonoids, coumarin, coumarins, biomarker, marker, markers, indice, indices, indicator, indicators, metabolite, concentration, exposure, exposed, serum, blood, plasma, urine, urinary, hair, nail, nails, toenail, toenails, erythrocyte, erythrocytes, RBC, adipose-tissue, thrombocyte, thrombocytes, skin, breath, saliva, faeces, feces, reliability, reproducibility, variability.