Universiteit Leiden

Master Computer Science

Medical Question Answering for Persian

Name:	Leila Darabi
Student ID:	S3385884
Date:	[27/02/2024]
Specialisation:	[Data Science]
1st supervisor:	Prof.Dr.Suzan Verberne
2nd supervisor:	Prof.Dr.Marco Spruit

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

This thesis examines the task of question answering within Persian medical question answering systems. The primary task is to retrieve relevant answers for each question using different models, with a particular focus on the Persian BERT model. At first our method involves identifying similar questions for each question to identify repeated questions. Users can then locate similar questions in the dataset, so it facilitates the retrieval of corresponding answers. Also, we use two evaluation strategies, strict and lenient, which help users to find the exact correct answer or similar ones. Furthermore, classification methods assist users to determine the appropriate category for each question, which improves the probability of finding the most relevant answer. Finally, we apply Named Entity Recognition (NER) to identify drug and disease names as entities within the dataset. Weighting these entities aims to improve the retrieval of answers across models. Interestingly, while our best model (TF-IDF) shows no significant response to weighted entities, our poorest model (ParsBERT) shows improved results.

Keywords: Question-Answering Retrieval; Persian Language; Medical Dataset; ParsBERT; TF-IDF.

Contents

Chapter 1

Introduction

In the digital era, individuals commonly pose inquiries on the internet through various websites. Numerous community question answering (CQA) forums, such as Quora[1] and Yahoo Answers[2], have gained popularity. These platforms serve as hubs where people post their questions and receive answers from the online community [1]. People frequently turn to these internet-based question-answering systems to find solutions to their queries, therefore improving these systems significantly influences the quality of information available to users. Generally, in these systems, identifying questions that are closely similar to the user's query makes it more straightforward for the system to locate the answer to the user's question. Subsequently, the next stage involves extracting the correct response to the question [1].

Within these CQA platforms, the lack of experts to address specific queries can lead to answers that may vary in accuracy. Consequently, the reliability of responses may differ, prompting users to exercise caution when considering the provided information.

Because user questions vary widely, covering everything from basic and common queries to very specific and unique ones, there is a growing interest in developing QA systems that are tailored to specific domains. In specific domain question answering systems, the accuracy of responses tends to be higher [2]. This is because questions posed by ordinary individuals are often addressed by experts, such as lawyers, doctors, and others with specialized knowledge in the relevant field.

Among specific domains, medical question answering has gained significant attention. More and more individuals are turning to these question-answering systems to find the information they need. This is because individuals often prefer to search and anonymously pose their questions on these platforms, facilitating the retrieval of accurate and informative answers [3].

Additionally, in these specific domains, the credibility of answers is often highest when provided by experts. For medical questions, in particular, answers generated by doctors carry a level of authority that individuals can trust with confidence. So retrieving answers in these systems is a more renowned approach.

While the majority of medical platforms operate in English, a significant challenge in this field is the scarcity of datasets. This issue becomes even more challenging for non-English speakers, as there are fewer datasets available and additional unique difficulties arise due to the specific characteristics of each non-English language and specific domains.

Among these non-English languages, Persian presents unique challenges such as right-to-left orientation and specific characters that pose difficulties in data preprocessing as well.

[1] https://www.quora.com/

[2] https://answers.yahoo.com/

Furthermore, the scarcity of non-English medical text on the internet adds another layer of challenge. While there is a broad spectrum of research in these areas, such as [4] for ParSQuAd (Persian Question Answering Dataset) and [5] providing ParsBERT, more research and attention are needed, especially in the field of medicine.
In this thesis, we compile a labeled dataset based on a crawl from a Persian website called hisalamat.com[3]. This platform enables users to anonymously ask questions to specialists, categorized by question categories. Users can subsequently follow up on their answers using the tracking number provided by the website.
Beyond the challenges that exist for Persian in this dataset, another notable characteristic is the length of questions submitted by laypeople. Typically, in question answering systems, questions are shorter than answers. However, in this context, questions from lay people tend to be longer, while medical doctors respond with comparatively shorter answers.
The difference in length between questions and answers can make the search for answers a bit tricky in this dataset. In other words, this difference can make it challenging to directly match up a user's question with the available answers. To navigate this challenge wisely, a recommended approach involves searching for questions in the dataset that are similar to the one the user has asked. By doing this, we aim to achieve two important goals. Firstly, it makes the process of finding relevant answers simpler by identifying questions that share similarities with the user's inquiry. Secondly, it enhances the user experience and knowledge about the question by offering access to answers that have already been provided for similar questions. This strategy not only helps bridge the gap between lengthy queries and concise responses but also makes the information retrieval process more efficient.
Moreover, since there are different categories in the dataset, it's beneficial to carefully retrieve the answers in the related categories. Doing this can improve the retrieval process. When users match their questions with specific categories, they can locate information more precisely and speed up the process of retrieving the answers that are directly related to their specific question. Also, since the dataset consists of medical terms, particularly drug and disease names, we expect that annotating these terms and incorporating them as entities within the system can influence the result of retrieving the correct answer.
In this research we address these questions:
Question I: How effective are various methods in identifying similar questions within our dataset?
Question II: To what extent are various methods effective in retrieving the relevant answer for each question?
Question III: To what extent can question classification into topical categories help the retrieval tasks?
Question IV: How can we enhance the performance of each model through re-ranking using medical terms?
This thesis consists of the following chapters: Chapter 2 provides the literature review and background of this research. In Chapter 3, we describe the data and preprocessing methods for this specific dataset, while in Chapter 4, the methodology and algorithms used in this research are described. The results and achievements of our research are shown in Chapter 5. Finally, in Chapter 6 and Chapter 7, we discuss our challenges and provide conclusions, respectively.

[3]https://www.hisalamat.com/

Chapter 2

Background

This chapter is divided into two main sections: Section 2.1 provides an overview of Question Answering (QA) and various methodologies, while Section 2.2 focuses on Medical Question Answering and its significance. To gain a comprehensive understanding, we explore Medical Question Answering in both English (Section 2.2.1) and Persian language (Section 2.2.2).

2.1 Question answering

Question answering (QA) is a prominent field in Natural Language Processing (NLP) that aims to develop automated systems capable of understanding and providing accurate answers to user questions [6]. As digital information continues to grow rapidly, QA systems play a crucial role in efficiently extracting specific knowledge from vast text collections. In 1999, the QA Track of the Text Retrieval Conference (TREC) initiated research on Question Answering from the standpoint of Information Retrieval (IR) [7]. In this context, QA systems provide answers to questions by retrieving concise text excerpts or phrases from numerous documents, potentially containing the answer itself. In that approach, QA is considered a fusion of IR and information extraction methodologies [8].
There are two main approaches to QA systems: text-based QA and knowledge-based QA. Knowledge-based QA relies on knowledge bases (KB) to find answers to user questions. Freebase, a well-known KB [9], has served as a benchmark in many recent studies on knowledge-based QA. KBs contain entities, relations, and facts. These facts are stored in a (subject, predicate, object) format, where the subject and object are entities, and the predicate represents the relation between them. The task involves two types of questions: single-relation and multi-relation questions. Simple questions are answered by a single fact in the KB, while multi-relation questions require reasoning over multiple facts in the KB [9]. In contrast, text-based QA retrieves answers to candidate questions by identifying the most similar answer text among candidate answer texts. Recent research has proposed various deep neural models for text-based QA, comparing question and candidate answers and producing a relevance score [10].
Traditional text based Question Answering Systems (QAS) can be categorized into three main modules: question processing, document and passage retrieval, and answer processing (answer extraction) [8].

1. Question processing: Question processing consists of two crucial steps: query formulation and answer type detection. During the query formulation step, question-processing converts a question into a search query by removing stop words and detecting keywords.

The query is generated based on query reformulation rules and usually represents a subset of the intended answer. In the answer type detection step, questions are categorized based on the type of answer they expect. For instance, there are some categories of questions, including those with Yes/No answers, factoid questions which consist of Wh-questions such as when, where, what and who, definition questions, list questions, and complex questions. [8] (Table 2.1 indicates examples of these question types and their expected answers). This is achieved using a classifier, which can be neural-based or feature-based, among other options [10].

Table 2.1: Example of question types and their expected answer

Question type	**Example of question**	**Expected answer**
Yes/No	"Is the Eiffel Tower located in Paris?"	Yes/No
Factoid	"Where is the Great Wall of China located?"	"China"
Definition	"What is the meaning of 'Photosynthesis'?"	Definition (The expected answer is a definition or explanation of the term 'Photosynthesis')
List	"Name five popular tourist destinations in France."	List (The expected answer is a list of tourist destinations.)
Complex	"What is the mechanism of action of a vaccine?"	information in a given context

2. Document and passage retrieval: The query generated in the previous step is then fed into an IR engine, which returns the top n retrieved documents. However, since answer extraction models typically operate on short segments of text, it is necessary to obtain concise sections of relevant text. The passage retrieval stage is a crucial part of the QA system because it identifies similar passages or sentences related to the input question. By doing so, it narrows down the search space and facilitates more accurate and efficient answer extraction [10].

3. Answer processing: Finally, in the last stage of the QA process, the most relevant answer is extracted from the selected passage. To ensure the accuracy and relevance of the response, an assessment of the answer's relevance to the input question is performed. This assessment helps determine how well the extracted answer aligns with the original question and is a crucial step in verifying the quality of the answer provided by the QA system [10].

In step two, estimating the relevance between question and the candidate answer passages is a critical and fundamental aspect of text-based QA systems, as it determines the relevance and accuracy of the responses provided.

2.1.1 Question answering methods

In the context of QA, there are essentially two approaches: extraction-based and generation-based methods. Generally, QA employs a query with the aim of searching through structured datasets to find the desired answer. Moreover, it can extend to encompass unstructured

natural language documents, from which relevant information can be extracted to provide the answer [11]. In generation-based question answering systems, which provided by [12], basically, the process begins with a question, and the system employs a classification step to determine its answerability. Once classified as answerable, the system proceeds to generate an informative natural language answer in response to the question [12].

More in general, QA is a synergistic approach that combines Information Retrieval (IR) and Natural Language Processing (NLP) with the aim of creating a system capable of automatically responding to human questions in a natural language [13]. Therefore, question answering stands out as a crucial and challenging task in the field of Natural Language Processing (NLP), centering on facilitating interactions between computers and human language [14]. Some key challenges in NLP for QA encompass speech recognition, information extraction [15], text summarization [16] and natural-language generation [17].

To comprehend the subject of question answering thoroughly, we need to define the relevant terms. A question phrase represents the segment of the question that specifies the sought information. Question type pertains to the categorization of the question based on its intended purpose. The term Answer Type refers to a class of objects that the question seeks to identify. Question Focus denotes the property or entity that the question aims to discover. Question Topic, on the other hand, pertains to the specific object or event that the question addresses. A candidate passage encompasses a wide range of content, varying from a single sentence to an entire document, retrieved by a search engine in response to a question. A candidate answer represents the text ranked according to its suitability as a potential response [18].

From another perspective, QA systems can be categorized into two main types based on their domain coverage: closed-domain and open-domain. Closed-domain QA systems are designed to handle questions within a specific domain such as medical or finance, leveraging domain-specific knowledge typically represented in ontologies. These systems only allow certain types of questions and provide short answers relevant to the specific domain [19]. On the other hand, Open-domain QA systems rely on general ontologies and world knowledge, enabling them to handle a wide range of questions. Moreover, these systems have access to a larger pool of data for answer extraction [20].

In their work, Yang et al. introduced the WikiQA dataset, which is designed for open domain question answering. They conducted evaluations using the WikiQA and QASent datasets, employing information retrieval-based models such as Word Count (Word Cnt), Weighted Word Count (Wgt Word Cnt), Learning Constrained Latent Representation (LCLR) [21]. The Word Cnt model functions by counting the occurrences of non-stop words in the question that also appear in the answer sentence. On the other hand, the Wgt Word Cnt model is similar to Word Cnt, but it takes into consideration the Inverse Document Frequency (IDF) weight of the question words, re-weighting the counts accordingly [22].

The advent of deep neural networks and after that the advent of BERT [23] has significantly transformed the landscape of question-answering (QA) methods in natural language processing. BERT demonstrated remarkable capabilities in understanding context and semantics. BERT has served as a cornerstone for many subsequent QA models. Most of the current state-of-the-art QA methods are BERT-based, capitalizing on BERT's pre-trained contextual embeddings and fine-tuning techniques. These models have done really well in many different question-answering tasks, like understanding passages and generating questions. What's interesting is that BERT, which was first trained in English, has inspired the creation of similar models in various other languages such as ParsBert [5] for Persian.

2.1.2 Sentence similarity

Semantic similarity refers to the measurement of how closely related two items are in terms of their meaning. These items could be concepts, sentences, or even entire documents. When applied to sentences or documents, it is specifically known as Semantic Textual Similarity (STS), a significant linguistic tool within Natural Language Processing (NLP). STS finds applications in various areas of NLP, including document classification, semantic search, information retrieval, question answering, sentiment analysis, and plagiarism detection [24].
In Natural Language Processing (NLP), multiple techniques exist for measuring sentence similarity. One such approach involves statistical analysis, where word frequencies within the corpus and documents are taken into account. An essential objective in corpus analysis is the computation of Term Frequency-Inverse Document Frequency (TF-IDF), a word weighting method used to assess the significance of words in the corpus [25]. Among the array of similarity measures, cosine similarity has emerged as a standout method and has been widely embraced by NLP researchers to this day [26]. Its effectiveness and versatility have made it a popular choice in various NLP applications, contributing to its continued prominence in the field.
Song et al. presented a method for measuring similarity between new questions and Frequently Asked Questions(FAQ) datasets. They used statistical measures based on dynamically formed vectors and semantic information based on WordNet and they could obtain reasonable results [27]. Ansari et al. used the Quora platform, a social media platform focused on Question and Answering (QA), to detect semantically duplicate questions. They employed various machine learning and deep learning techniques, including feature engineering, feature importance analysis, and extensive experimentation. After evaluating multiple models, they determined that the XGBoost model, leveraging character-level term frequency and inverse term frequency, yielded the most promising results compared to other models [6].
Following the introduction of BERT by Devlin [23], SBERT [28] was developed with a specific focus on encoding sentences and text, enabling more effective comparison of sentence and text segment similarity. SBERT employs a Siamese or twin network architecture, which results in fixed-length sentence embeddings. These embeddings are versatile and find application in numerous NLP tasks, such as sentence similarity, semantic search, and IR. While SBERT uses Bi-Encoders, which generate sentence embeddings independently for each sentence, BERT uses the Cross-Encoder approach. In the case of Bi-Encoders, individual sentences A and B pass through BERT separately, resulting in two separate embeddings vectors, 'u' and 'v'. These embeddings are then used to compute cosine similarity to measure the similarity between two sentences. Cross-Encoders process pairs of text jointly, concatinating them instead of individual sentences. These text pairs can represent various forms, such as question-answer pairs and sentence pairs used in paraphrase detection. After processing the input pairs, Cross-Encoders produce a representation or similarity score that quantifies the similarity or relationship between the two text segments. This approach, emphasizing the contextual interaction between text pairs, offers valuable insights in understanding the relationships between textual content. Figure 2.1 illustrates the distinction between the bi-encoder and Cross-encoder models [28].
In this thesis, we will address both similar question identification and answer retrieval.

2.2 Medical question answering

Question Answering Systems (QAS) have remarkably improved recently but the medical field has received relatively limited attention in research due to several challenges. These challenges

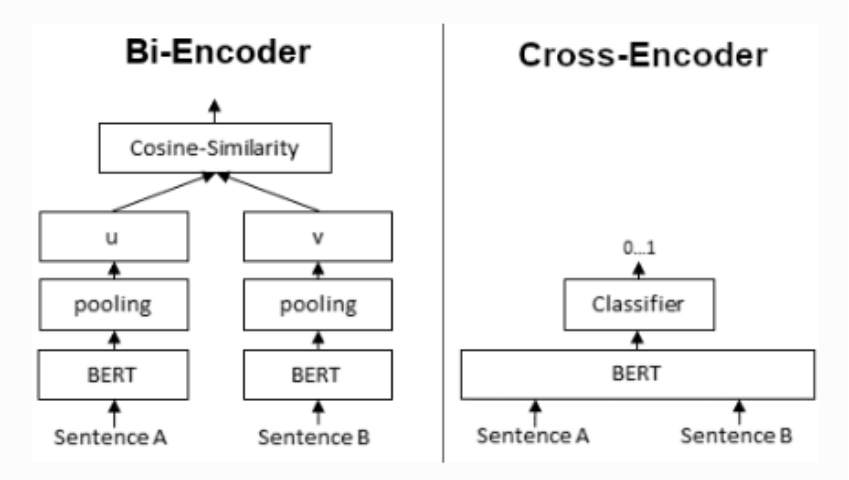


Figure 2.1: Bi-Encoder vs. Cross-Encoder [28]

include the scarcity of labeled data for medical texts, the diverse nature of medical questions, and the limitations of models in comprehending complex question-answer texts [29]. Indeed, emerging deep learning techniques in this area hold great promise and assist for successfully tackling medical tasks [30]. With the increasing reliance on online platforms for health-related inquiries, there is a growing demand for efficient retrieval of similar questions and their corresponding answers. This benefits users in finding relevant information easily and supports medical websites in handling inquiries effectively. Researchers are actively exploring this field, including in languages like English and Persian, to enhance the capabilities of question answering systems.

2.2.1 Medical question-answering for English

In recent years, QA systems have gained significant attention in the medical field, playing a vital role in providing accurate and timely information. Interestingly, more and more people are now turning to the Internet to learn about their health conditions [3].
A scientific research in USA shows that "*69% of U.S. adults track a health indicator like weight, diet, exercise routine, or symptom. Of those, half track "in their heads," one-third keep notes on paper, and one in five use technology to keep tabs on their health status*" [31]. This means the trend of seeking medical advice online continues to grow. Thus, the importance of medical QA systems becomes even more pronounced. These systems hold the potential to assist healthcare professionals, researchers, and patients alike, granting them access to precise medical information. Even physicians can use these systems to help them with difficult medical decisions that come up while treating a patient. This, in turn, leads to better healthcare services [32]. So, the future of healthcare is becoming increasingly reliant on these advanced QA systems and the valuable insights they offer.
Traditionally, meeting information needs has been accomplished through the use of Information Retrieval (IR) systems, however, classical IR still faces challenges. In contrast, Question Answering (QA) systems offer a more straightforward and intuitive approach. Instead of providing users with a list of relevant documents to read, QA systems directly furnish answers to users' questions [3]. Thanks to the swift advancement of computing hardware, contemporary QA models, particularly those leveraging deep learning [33, 34], now demonstrate comparable performance on numerous benchmark datasets. As a result, these modern QA models have found

successful applications in general domain search engines and conversational assistants [35].
In the realm of medical and bio question answering, there has been a shift from traditional IR-based methods to the adoption of state-of-the-art transformer-based models. Various classic bio QA systems, including EPoCare [36], PICO and knowledge-extraction-based BQA systems [37, 38], MedQA [39], Health on the Net QA [40], AskHERMES [41], and more, have been developed and utilized.
A pivotal challenge in this field is BioASQ [42], which has been conducted annually since 2013 to evaluate biomedical natural language understanding systems. Over the years, numerous BQA systems have been proposed and tested in BioASQ, leading to remarkable performance improvements. For instance, QA performance has risen from approximately 20% in top factoid mean reciprocal rank and list F-measure in BioASQ 1 to around 50% in BioASQ 8 [43]. The introduction of transformer and BERT [23] has further boosted advancements, with nearly all top-performing BQA systems incorporating biomedical pre-trained language models (PLMs), such as BioBERT [44], in their systems. This integration has played a significant role in achieving more accurate and effective bio QA results.
Like other QA systems, medical QA systems are typically organized into three main modules: question processing, document and passage retrieval, and answer processing [8]. These modules work together to process user questions, retrieve relevant medical information from documents, and extract accurate answers. Additionally, there are various types of medical QA systems and numerous authors have put forward diverse methods for medical question answering systems (QASs). Sharma et al. have presented a comprehensive and in-depth analysis of various question answering systems, including a comparison of four biomedical systems: AskHermes, MedQA, HONQA, and MiPACQ [45]. Furthermore, SemBioNLQA, introduced by Sarrouti et al., is a biomedical question answering system. This comprehensive system comprises question classification, document retrieval, passage retrieval, and answer extraction modules. By accepting questions in natural language format, SemBioNLQA generates concise and accurate answers, and it also provides a summary of the results [46]. Athenikos, Han, and Brooks [47] employ description logic, specifically OWL description logic, to address medical questions. They extract keywords and topics from the questions to match predefined answering logics. However, the reliance on manual labor to define these logics restricts the applicability of their approach to a broader range of question answering tasks.
Despite advancements in KBQA (Knowledge-Based Question Answering), there are still existing challenges. Traditional KBQA approaches have limitations due to their dependence on historical cases, which may not cover all possible scenarios. Xiaofeng Huang presents a novel approach for medical domain Question Answering (KGQA) using a knowledge graph. The approach involves constructing a medical knowledge graph by extracting named entities and their relations from medical documents. It then uses key information and question intentions extracted from the question to score related entities using an inference method based on weighted path ranking. The approach demonstrates efficiency in understanding questions, connecting them to the knowledge graph, and inferring answers, as supported by theoretical analysis and real-life experimental results [48]. Furthermore, researchers applied their Question Answering models to various datasets. One notable example is the work by Timo Moller et al. introduced COVID-QA, a comprehensive Question Answering dataset comprising 2,019 QA pairs specifically focused on COVID-19. The dataset was meticulously annotated, and the researchers conducted an evaluation using a fine-tuned RoBERTa base model trained on both the SQuAD dataset and the COVID-QA dataset. The results revealed significant improvements in performance compared to using the model trained solely on SQuAD, thereby emphasizing the effectiveness

of incorporating the COVID-QA dataset into the training process [49].
In recent research using Large Language Models (LLMs) for medical question answering, Liévin et al. in their paper titled "Can large language models reason about medical questions?" talked about the use of Chain-of-Thought (CoT) with GPT-3.5 models for medical question answering. They investigate various CoT prompting variations, including zero-shot and domain-specific cues, and evaluate performance and limitations through expert assessment. Additionally, by leveraging the Codex beta program, the paper demonstrates that scaling inference-time compute with Codex allows for 5-shot CoTs to do medical question answering tasks, showcasing the potential of large language models in the medical domain [50].

2.2.2 Question-answering for Persian

Researching different languages other than English in the field of NLP, particularly in question answering, has emerged as a novel and challenging area of research. For example, the creation of the Chinese Question Answering System in the Medical Domain (CQASMD) aims to furnish users with medical information. It uses a large medical knowledge base and advanced models such as FastText and Word2Vec to classify and process users' questions, matching them with the most similar question and providing relevant answers from the medical knowledge base. This system serves as a valuable tool for self-disease diagnosis and treatment [51].
Developing NLP systems, especially for question answering, in Persian language is a tough challenge. Persian's distinct alphabet and linguistic structure require specialized datasets and models designed to suit its unique features. Furthermore, creating efficient QA systems for medical Persian language presents distinct challenges due to the intricate nature of medical terminology and the nuances specific to the Persian language. As Persian is widely spoken in the Middle East, including countries like Iran, Tajikistan, and Afghanistan, it requires specialized approaches and dedicated resources to effectively address its linguistic complexities within the domain of medical question answering.
As a result, researchers in this area have taken initiatives to create Persian datasets, such as PerCQA, which is the Persian Community Question Answering Dataset [52], and PQuAD, which is a Persian question answering dataset [53]. They have also developed Persian NLP libraries, such as Hazm [54], a comprehensive Persian NLP library. Moreover, to leverage state-of-the-art methods like transformers, ParsBERT [5] was introduced, based on BERT [23], specifically tailored for the Persian language. These advancements are crucial steps towards overcoming the challenges and unlocking the full potential of NLP in Persian, benefiting various applications such as QA and other language-related tasks.
Additionally, for evaluating Persian web-based Question Answering Systems, Tohidi et al. introduced a Multi-Objective Evolutionary algorithm (MOEA) called NSGA-II. The evaluation of the system using standard and web data demonstrates significant and effective results when compared to other existing systems [55]. In the domain of Medical Question Answering, research in the Persian language is limited due to the scarcity of Persian documents and websites. However, Veisi and Shandi provided a medical question answering system in the Persian language with three main modules: question processing, document retrieval, and answer extraction. The system achieved an accuracy of 83.6% in answering questions about diseases and drugs, employing customized language processing tools and similarity detection algorithms [2].
Taghizadeh et al. introduced SINA-BERT, a language model built on BERT [23], was introduced to fill the gap of a reliable Persian language model in the medical field. It underwent pre-training

on a vast collection of medical content from formal and informal sources on the internet to enhance its performance on health-related tasks. SINA-BERT was applied to tasks like medical question categorization, medical sentiment analysis, and medical question retrieval, and with its consistent architecture across these tasks, it demonstrated superior performance compared to earlier BERT-based models available in Persian [56].

This thesis focuses first on collecting a dataset for Persian medical Question Answering. The study further involves the development and assessment of techniques for identifying similar questions and retrieving appropriate answers, leveraging the newly constructed Persian medical QA dataset. The implementation and evaluation of these methods are key components of this research, aiming to enhance the efficiency and accuracy of medical QA systems in the Persian language context.

Chapter 3

Data

In this Chapter, we closely examine the dataset, its preprocessing steps, and the challenges involved. Additionally, we describe the importance of relevance assessment and annotating data for this dataset. We annotate the data twice: once for similar questions and another time for similar answers. This prepares the dataset for further examination.

3.1 Data

We collected data from a Persian medical website named hisalamat.com[1]. This website offers various features such as a directory of physicians, an "Ask Your Question" section, "medical content", a "health magazine", and a "health radio". The web site provides information about physicians and their specialties. Users can anonymously ask their medical questions, categorized by different medical groups. Patients are also given a question tracking code to easily search for their own questions. When a question is posted, doctors can read and answer it based on their area of expertise. Additionally, users can search for specific keywords to find related questions or create new ones. Only one doctor can answer each question, and their name and specialty are recorded in the answer section.
To extract and crawl the web to collect data from websites, we used Beautiful Soup[2], then data is saved in a pandas data frame. The dataset has 1,087 records with four fields. The "Title" holds the questions, while the "Body" contains the answers. The "Group" categorizes the questions by medical topics, and the "Name of the Doctor" shows who answered each question. The "Date" of the questions is provided in the website link, and we sorted the index according to the date. Since the questions are asked anonymously, there are no names of patients in the dataset. However, each patient is assigned a question tracking code for easy monitoring and follow-up.
There are 36 distinct groups, with each group dedicated to related questions. This design ensures that the doctor who responds to questions within a specific group is knowledgeable and specialized in the corresponding medical topic.
Remarkably, there are 84 doctors who contribute their expertise to answer these questions. Figure 3.1 and Table 3.1 illustrate the original and translated data, respectively.

[1] https://www.hisalamat.com/
[2] https://pypi.org/project/beautifulsoup4/

IndexNumber	doctor	group	body	title
7	دکتر سیّد مصطفی لنگری	جراح عمومی	خیر	عمل مامو پلاسي انجام میدید اگر انجام میدید قیمت بفرمایید
8	گفتار درمانی درخشان	گفتار درمانی	در این حیطه همکاران ادیولوژیست بهتر میتونن نظر بدن	اگه کم شنوایی حسی عصبی باشه احتمال داره ک بعد از ب دنیا اومدن شروع ب رشد کنه و ب مرور زمان بهتر بشه نسبت ب قبل؟
9	دکتر سیّد مصطفی لنگری	جراح عمومی	احتمالا فیستول یا ابسه انال دارید که نیاز به عمل جراحی دارد	آقایی هستم 30 ساله ، در قسمت خروجی مقعد (سوراخ) چیزی شبیه به جوش زده ، متورم هست و درد هم داره برای اجابت مزاج با درد مواجه هستم ولی جوش نیست چون اصلا هیچ علائمی شبیه به جوش نداره ممنون میشم راهنماییم کنید در ضمن خونریزی هم نداره فقط قمز شده متورم و همراه با درد
10	دکتر شهرزاد جوان	زنان و زایمان	از کیت تخمک گذاری استفاده کنید یا به پزشک مراجعه کنید	من سی و هشت هفته زایمان کردم بچم مرده بود دوباره میخوام سریع حامله شم از کجا روز تخمک گذاری مو بفهمم اون یکی پسر بود میخوام اینم پسر باشه
11	گفتار درمانی درخشان	گفتار درمانی	من تخصصی در حیطه گفتاردرمانی کار میکنم شما باید با همکاران روانشناس و روانپزشک در این حیطه مشورت کنید	من با مردی ازدواج کردم که یه دختره 17 ساله داره که با ما زندگی میکنه بچه7ماهه بدنیا اومده الان که با مازندگی میکنه من متوجه شدم با بقیه بچه ها فرق داره تا باهاش حرف نزنی حرف نمیزنه اصلا ظاهرش براش مهم نیست هرچی بهش میگم هر کی هرچی بهت گفت بیا بگو بازم نمیگه حتی جلسات و امتحانات مدرسه رو نمیگه هر کی هر کاری یا رفتاری باهاش میکنه از خودش دفاع نمیکنه فقط نگاه میکنه و جدیدا دروغ هم میگه من فک کنم این از بچه گیش یه مشکلی داشته که بزرگتر هاش پیگیری نکردن اینم همین جوری مونده چیکار کنم خوب شه پیشه چه پزشکی بهرمش؟

Figure 3.1: Showing examples from Persian dataset

Table 3.1: Translations of the examples in Figure 3.1 to English

Inx	Title	body	Group	Doctor
7	Did you do Mammoplasty, if you do, tell me the price?	NO	General surgery	Dr.Seyed Mustafa Langari
8	If hearing loss is sensorineural, is it possible that it will start to grow after birth and get better over time?	In this area, audiologist colleagues can give a better opinion	Speech Therapy	Derakhshan speech therapy
9	I am a 30-year-old man, there is something similar to a boil in the exit part of the anus (hole), it is swollen and it hurts. I am facing pain for defecation.But it's not a pimple because it doesn't have any symptoms similar to a pimple at all. Thank you for your help. Besides, it's not bleeding, it's just swollen and with pain.	You probably have an anal fistula or abscess that needs surgery	General surgery	Dr.Seyed Mustafa Langari
10	I gave birth at 38 weeks and my baby died. I want to get pregnant again quickly. How can I know on the day of ovulation that one was a boy? I want this one to be a boy.	Use an ovulation kit or see a doctor.	Women's medicine	Dr.Shahrazad Jovan
11	I am married to a man who has a 17-year-old daughter who lives with us. A 7-month-old baby was born.Now that she lives with us, I noticed that she is different from other children, she doesn't talk unless you talk to her. she does not care about her appearance at all.Whatever I tell her , whatever he told you, come and tell me, she won't say it again.She doesn't even tell about school meetings and exams.Anyone who does anything or behaves with her is not defending herself, she is just watching.And now she is lying.I think this kid had a problem that her elders did not follow up.What should I do to get better, what kind of doctor should I take?	I specialize in speech therapy. You should consult with psychologist and psychiatrist colleagues in this area.	Speech Therapy	Derakhshan speech therapy

In this dataset, questions have varying lengths and different numbers of words. Some of the questions are quite short, while others are longer with extensive descriptions. The longest question has 3264 characters with 707 words, and the shortest one has 16 characters with 2 words. For answers, most of them are shorter than the questions, but the longest answer has 2323 characters with 429 words, while the shortest one consists of only 2 characters, which are two spaces, indicating no words. In total, the average length of the questions is 295 characters with 59.93 words, and the average length of the answers is 160 characters with 28.66 words.

3.2 Preprocessing

Since the data is in Persian, our approach and the Python library for preprocessing are specifically tailored for Persian. We used "HAZM" [57] which is a Python library designed for carrying out natural language processing operations on Persian text. It provides a wide range of functionalities for examining, handling, and comprehending Persian textual content. With Hazm, we have the capability to normalize text, tokenize sentences and words, perform lemmatization of words, ascribe part-of-speech (POS) tags, recognize dependency relationships, generate word and sentence embeddings, and even access well-known Persian language collections.
Given that the data originates from a Persian medical question answering context, **one challenge** is that the data is not entirely clean. In addition to common Persian stopwords, it contains numerous unwanted words and phrases. For instance, for each question and answer, there are "Question :" and "Answer :" words, which are unwanted. Also, phrases such as "Hi", "How are you", "I have a question", "Thank you","Dear lovely doctor" and others of a similar nature. Having irrelevant words in the dataset can negatively impact the performance of the model, particularly for TF-IDF. Thus, removing them can help us to improve the accuracy and efficiency of models. Figure 3.2 indicates some of these phrases which we carefully remove from the dataset. Therefore, we manually curated the data and improved its quality by editing the questions and answer texts, removing all non-related utterances from the texts. Table 3.2 illustrates part of these words and phrases along with their translations.
Another challenge that we faced pertained to the number of categories. As explained in Section 3.1, our dataset comprises 36 categories. Due to the imbalance in the number of questions across these categories, we decided to merge certain categories that have less data with others that have more data and are also highly similar to each other. One example is the "Kidney and urinary tract surgery" category, which has just one question and answer. Since it is similar to the "Kidney and urinary tract" category with 42 questions, we decided to merge them. Another example is the merging of "Obstetrics and Gynecology", "Obstetrics and Gynecology specialist", and "Midwifery" categories into a single category. Thus reducing the total number of categories from 36 to 25. Although we attempted to merge small categories with similar ones, some categories could not be merged with others due to dissimilarity. Therefore, we opted to retain them with their respective amount of data. The smallest category after merging has 3 questions, which is the "Laboratory sciences" category. We will use the existing categories to classify questions based on their respective categories, which will be introduced in Section 5.3.

Question	Answer
پرسش:سلام دکتر خسته نباشید میخواستم بدونم که آیا میشه در زمان ابتلا به سرماخوردگی واکسن آنفولانزا تزریق کرد؟	**پاسخ** واکسن باید در سلمت کامل بعد از بهبودی از سرما خوردگی تزریق شود
پرسش:سلام خسته نباشید من پریود های منظم و دقیقی دارم این ماهی برای بارداری اقدام کرده بودم ۲ روز از موعدم گذشت ازمایش خون دادم منفی بود حالت تهوع دارم طرف صبح امکانش هست که در ازمایش خطا رخداده باشد بدلیل اینکه زود رفتم ؟ اگر احتمال خطا نیس چرا پریود نشدم؟؟	**پاسخ** سلام احتمال حاملگی کمه ولی برای اطمینان یک هفته بعد در صورت پریود نشدن مجددا آزمایش دهید
پرسش:سلام خسته نباشین.ببخشید من بخاطر نازایی ماه قبل به پزشک مراجعه کردم.بعد از سونو گفتن اندازه فولیکولت کوچیکه یه سری قرص بهم دادن و قرار شد این ماه روز سوم قاعدگی دوباره برم مطب و قرار بود بهم امپول تجویز کنند .اما متاسفانه امروز که روز سوم پریودم بود رفتم مطب بسته بود.ایا حتما امروز باید آمپول تجویز میشد یا اینکه میتونم روز پنجم پریودم برم؟	**پاسخ** تا روز ۵وقت دارین
پرسش:سلام به شما دکتر مهربان ودوست داشتنی،حدودا یک ماه پیش فیزیوتراپی گردن و کمر پیش شما انجام دادم ودو جلسه لیزر خیلی بهتر شدم؟؟؟؟،درطول مدت درمان دکترم قرص های غضروف ساز و تقویتی تجویز کرد وتمام کردم الان 25روزیه تمام شده،درد شدید درانگشتان دست ومچ دست وپا اذیتم میکنه؟؟؟؟،کم کاری تیروئید دارم وچند وقته ضعف و ناتوانی تمام جسمم رافراگرفته،،؟؟؟؟باکمک کف دست میخوام بلند شم دستم تحمل نداره وجاخالی میده ومیفتم وچند وقته از زانو به پایین انگار مایع گرم ریختن روپام،براچند لحظه احساس گرما میکنم،همه فکر میکنن ادا در میارم،من قبلا خیلی پر جنب و جوش بودم ولی الان اضافه وزن دارم بعداز کمی حرکت داخل خانه حتما باید دراز بکشم بعد دراز کشیدنم صدا وجابجا شدن مهرها راحس میکنم وارام ،ممنون میشم کمکم کنید؟؟؟؟؟؟؟	**پاسخ** **سلام دوست گرامی**، لطفا برای مشاوره به صورت حضوری مراجعه بفرمایید.

Figure 3.2: Showing examples of removed words and phrases

3.3 Relevance assessment

Relevance assessment plays a crucial role in information retrieval research and is fundamental for improving the accuracy and efficiency of search engines. In relevance assessment, human assessors or judges evaluate the retrieved documents and assess their relevance to a given query. The documents are typically categorized into different relevance levels, such as highly relevant, somewhat relevant, or not relevant, based on the criteria. These assessments are used to create relevance judgments or ground truth data, which can be used to measure the effectiveness of search systems through metrics like precision, recall, and F1 score [58].

There are several Python libraries and tools that can be used for performing relevance assessment in the context of information retrieval and search engine evaluation. Some of these libraries and tools include:

TREC Eval[3]: TREC (Text Retrieval Conference) provides evaluation frameworks and datasets for information retrieval research. TREC Eval is a widely used tool for evaluating the effectiveness of information retrieval systems. It calculates various evaluation metrics based on relevance judgments and retrieved documents.

pytrec_eval: This is a Python library for working with TREC evaluation. It allows you to compute common evaluation metrics like precision, recall, F1 score, mean average precision, and more using relevance judgments [59].

[3]https://trec.nist.gov/trec_eval/

Table 3.2: Showing examples of removed words and phrases with their translation

English	Persian
Hello, dear questioner	با سلام خدمت شما پرسش کننده عزیز
Hello dear friend	با سلام خدمت شما دوست عزیز
Hello again, in continuation of the previous question	با سلام دوباره در ادامه سوال قبلی
Hello dear friend	با سلام دوست عزیز
Greetings, your prayers and prayers are also accepted	با سلام طاعات و عبادات شما هم قبول
Greetings and Regards	با سلام و احترام
Greetings dear mother	با سلام و احترام خدمت مادر عزیز
Greetings and Regards. Lady doctor.	با سلام و احترام. خانم دکتر.
Greetings and courtesy	با سلام و ادب
Hello and thanks for the advice of the health department staff	با سلام و تشکر از مشاوره کارکنان بخش سلامت
Hello and goodbye	با سلام و خداقوت
Hello and don't be tired	با سلام و خسته نباشی
greetings	با سلام و خسته نباشید
Hello and don't be tired, doctor	با سلام و خسته نباشید اقا دکتر
Hello and don't be tired, Mr. Doctor	با سلام و خسته نباشید خدمت آقای دکتر
greetings.	با سلام و خسته نباشید.
Greetings	با سلام و درود
Hello and peace be upon you	با سلام و درود بر شما

In this study, based on TREC Eva, we used *pytrec_eval* library. To do this, we need to create qrel files containing the relevance labels for question-answer pairs. In TREC evaluation, the qrel (relevance judgment) file and the output of a retrieval model have specific formats. qrel file format (Ground Truth) has four main fields:

- Query ID: This field represents the identifier or number associated with a query. Each query should have a unique ID.
- Q0: This is a constant field typically set to "q0". It's used to separate the query ID from the document ID.
- Document ID: This field contains the identifier of the document being evaluated.
- Label: The label represents the relevance score of the document for the given query. The relevance level is usually assigned as an integer value [59] [60].

To evaluate our models in this study, we used the qrel file as the ground truth. Given that we applied both Question-Question similarity (Section 5.1) and Question-Answer retrieval (Section 5.2), we applied evaluations using distinct labeling criteria. For question-question similarity, which was manually labeled, the label values were defined as 0 (not relevant), 1 (partially relevant), and 2 (very relevant). In the case of question-answer retrieval, the labels were simply 0 (not relevant) and 1 (relevant).

An output file of model has the below format:

- Query ID: Similar to the qrel file, this field represents the identifier of the query that generated the result.
- Q0: As in the qrel file, this is a constant field, typically set to "q0", used to separate the query ID from the document ID.

- Document ID: This field contains the identifier of the document that was retrieved.
- Rank: The rank indicates the position of the retrieved document in the ranked list. The document with the highest rank is considered the most relevant.
- Score: The score is a numerical value that represents the relevance score assigned by the retrieval model to the document for the query.
- Explainability (XP): This field may be used to provide additional information or explanation about the ranking, but it's not a standard field and is often optional [59][60].

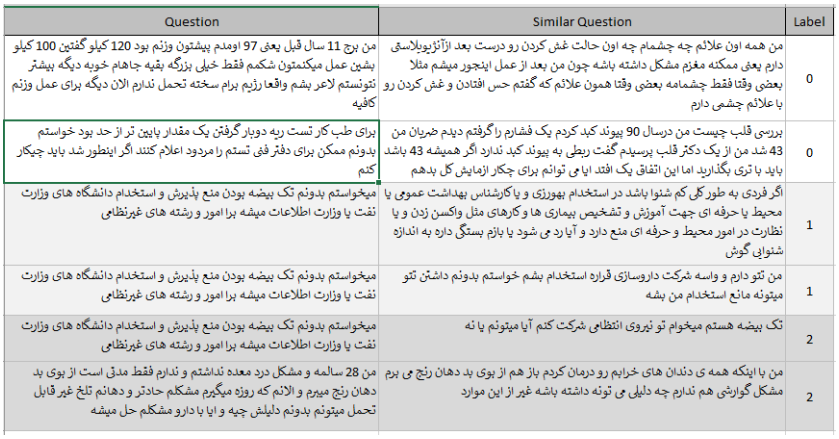
In our studied dataset, there is one correct answer for each question provided by medical experts (Doctors). For the initial model evaluation, we employ *strict evaluation* criteria. we assume that the qrel file is exactly equivalent to the correct answers provided by the experts and already present in the dataset. Therefore, our qrel file has 1087 rows, and each answer is labeled as 1 as relevant, while the other answers are assumed to be irrelevant.
To enhance the performance of our models, we employed another method of evaluation, *lenient evaluation*. In this approach, we consider that each answer may have a similar pair. Consequently, we identified the top 5 similar answers for each correct answer consisting of the exact correct answer.
As we explained in Section 3.4.2, since TF-IDF has better results in finding answer similarity, we extracted four similar answers using TF-IDF for each correct answer. Recognizing that each answer may not exhibit exact similarity, we manually annotated the file, labeling all similar answers as relevant with a value of 1. This ensures we have at least one correct answer identified for each query, and ideally, at least five answers in total.
With these formats and strategies for ground truth and the output of the model, we compute measurements such as MAP, MRR, and Recall@k (k = 5, 50, 100 and 200) for each of the selected models to evaluate their performance.

3.4 Annotating data

Annotating and labeling data plays a significant role in various supervised machine learning methods, such as computer vision, classification, and more. Moreover, it serves as the foundation for a wide range of NLP tasks. Annotating and labeling data in NLP is essential and challenging [61].
In this study, we performed two annotation tasks: one to identify similarity between questions and a second one to determine similarity between the correct answers in our dataset and other answers.

3.4.1 Annotating similar questions

We employed TF-IDF and ParsBERT models, with the latter not being pretrained on our dataset. To enhance the accuracy of both BERT and TF-IDF models, we performed labeling. The reason behind annotating the data is: firstly, it significantly impacts the evaluation and testing process, allowing us to assess our models by comparing their results with the annotated data. Secondly, the specificity of the domain is another driving factor for data annotation in this study. Given that we used a Medical Question Answering dataset, precise labeling of medical questions is necessary due to the domain's specialized terminology and disease names.



Question	Similar Question	Label
من برج 11 سال قبل یعنی 97 اومدم پیشتون وزنم بود 120 کیلو گفتین 100 کیلو بشین عمل میکنمتون شکمم فقط خیلی بزرگه بقیه جاهام خوبه دیگه بیشتر نتونستم لاغر بشم واقعا رژیم برام سخته تحمل ندارم الان دیگه برای عمل وزنم کافیه	من همه اون علائم چه چشمام چه اون حالت غش کردن رو درست بعد ازآنژیوپلاستی دارم یعنی ممکنه مغزم مشکل داشته باشه چون من بعد از عمل اینجور میشم مثلا بعضی وقتا فقط چشمامه بعضی وقتا همون علائم که گفتم حس افتادن و غش کردن رو با علائم چشمی دارم	0
برای طب کار تست ریه دوبار گرفتن یک مقدار پایین تر از حد بود خواستم بدونم ممکن برای دفترفنی تستم را مردود اعلام کنند اگر اینطور شد باید چیکار کنم	بررسی قلب چیست من درسال 90 پیوند کبد کردم یک فشارم را گرفتم دیدم ضربان من 43 شد من از یک دکتر قلب پرسیدم گفت ربطی به پیوند کبد ندارد اگر همیشه 43 باشد باید با تری بگذارید اما این اتفاق یک افتد ایا می توانم برای چکار ازمایش کل بدهم	0
میخواستم بدونم تک بیضه بودن منع پذیرش و استخدام دانشگاه های وزارت نفت یا وزارت اطلاعات میشه برا امور و رشته های غیرنظامی	اگر فردی به طور کل کم شنوا باشد در استخدام بهورزی و یا کارشناس بهداشت عمومی یا محیط یا حرفه ای جهت آموزش و تشخیص بیماری ها و کارهای مثل واکسن زدن و یا نظارت در امور محیط و حرفه ای منع دارد و آیا رد می شود یا بازم بستگی داره به اندازه شنوایی گوش	1
میخواستم بدونم تک بیضه بودن منع پذیرش و استخدام دانشگاه های وزارت نفت یا وزارت اطلاعات میشه برا امور و رشته های غیرنظامی	من تتو دارم و واسه شرکت داروسازی قراره استخدام بشم خواستم بدونم داشتن تتو میتونه مانع استخدام من بشه	1
میخواستم بدونم تک بیضه بودن منع پذیرش و استخدام دانشگاه های وزارت نفت یا وزارت اطلاعات میشه برا امور و رشته های غیرنظامی	تک بیضه هستم میخوام تو نیروی انتظامی شرکت کنم آیا میتونم یا نه	2
من 28 سالمه و مشکل درد معده نداشتم و ندارم فقط مدتی است از بوی بد دهان رنج میبرم و الانم که روزه میگیرم مشکلم حادتر و دهانم تلخ غیر قابل تحمل میتونم بدونم دلیلش چیه و ایا با دارو مشکلم حل میشه	من با اینکه همه ی دندان های خرابم رو درمان کردم باز هم از بوی بد دهان رنج می برم مشکل گوارشی هم ندارم چه دلیلی می تونه داشته باشه غیر از این موارد	2

Figure 3.3: Showing examples of different question-question similarity labels

Table 3.3: Showing the translation examples of different question-question similarity labels

Question	**Similar Question**	Label
I came to you 11 years ago, in 1397, my weight was 120 kilos, you told me to sit at 100 kilos, I am going to operate on you, my stomach is just too big, the rest of my body is fine, I couldn't lose weight any more, I can't bear the diet, now my weight is enough for the operation.	I have all those symptoms, whether it's my eyes or that feeling of fainting right after angioplasty, which means my brain might have a problem because I get like this after the operation, for example, sometimes it's just my eyes, sometimes the same symptoms I said I feel like falling and fainting with eye symptoms	0
For medical work, taking the lung test twice was a value lower than the limit. I wanted to know that the technical office might declare my test as failed. If that happens, what should I do?	What is a heart check? I had a liver transplant in 1390. I took my blood pressure and saw that my heart rate was 43. I asked a cardiologist. He said that it has nothing to do with liver transplant. If it is always 43, you should put it with Terry, but this happens once. Can I have a full test? to give	0
I wanted to know if having one testicle is prohibited for admission and employment in the universities of the Ministry of Oil or the Ministry of Information for civilian affairs and fields.	If a person is generally hard of hearing, it is prohibited to hire a physiotherapist or a public or environmental or professional health expert for training and diagnosis of diseases and work such as vaccination or supervision in environmental and professional affairs. It depends on the hearing size of the ear.	1
I wanted to know if having one testicle is prohibited for admission and employment in the universities of the Ministry of Oil or the Ministry of Information for civilian affairs and fields.	I have a tattoo and I am going to be employed by a pharmaceutical company. I wanted to know if having a tattoo can prevent me from being hired.	1
I wanted to know if having one testicle is prohibited for admission and employment in the universities of the Ministry of Oil or the Ministry of Information for civilian affairs and fields.	I am single testicle, I want to join the police force, can I or not?	2
I am 28 years old and I have never had a problem with stomach pain and I don't have it. I have been suffering from bad breath for some time and now that I am fasting, my problem is more acute and my mouth is unbearably bitter. May I know what is the reason and whether my problem can be solved with medicine.	Even though I treated all my bad teeth, I still suffer from bad breath, I don't have digestive problems, what could be the reason other than these things.	2

To conduct the annotation, we employed a manual process. We divided the data into two parts: part one consisting of 869 records (80 percent of the data) and part two consisting of 218 records (20 percent of the data). Subsequently, we identified the top 5 similar questions for each question in part two from part one. To to this, we used three different models,TF-IDF, ParsBERT, and a variation of ParsBERT[4]. This resulted in a maximum of 15 similar questions for each original question. Subsequently, we merged all the results from each model and created a pooled dataset. Now this pooled dataset consists of 3,270 records. Since some of the similar questions extracted by each model were identical, we removed these duplicates, resulting in 2,889 unique records across the three models. Irrespective the similarity scores assigned by the model that extracted them, we annotated the questions manually, carefully reviewing each question and its corresponding similar questions. We assigned labels as follows: 0 for not similar, 1 for partially similar, and 2 for very similar. In this context, similarity means that two questions have the same meaning and dissimilarity indicates that they do not have the same meaning. Figure 3.3 and Table 3.3 provide illustrations of examples for each category and their respective translations.
The number of question-question pairs annotated is illustrated in Table 3.4. Finally, we used this annotated dataset as a qrel (ground truth) to perform relevance assessments.

Table 3.4: The number of question-question pairs annotated for three models

Label	**The number of question-question pairs**
0 - not similar	1755
1 - partially similar	750
2 - very similar	385

3.4.2 Annotating similar answers

The second time of annotating the dataset was conducted to identify similar answers for each correct answer in our dataset, thereby expanding the relevance (ground truth). Given that there is only one correct answer available for each question, relying only on this answer for model retrieval results in low accuracy. Therefore, we selected to broaden the range of acceptable answers to include those that are similar. To achieve this, since TF-IDF got better result to find similarity between sentences in this dataset (Section 5.1, Table 5.1), we decided to use TF-IDF to find the similarity between answer-answers. Examining the results, which we will see in Chapter 4, we found that TF-IDF outperformed BERT in terms of semantic similarity, likely attributable to the presence of specific terms within our dataset. Thus, we chose the TF-IDF model to identify similar answers. We discovered four answers similar to each correct answer and carefully annotated them. In the creation of the qrel file, we appended these four similar answers to the exact correct answer. As a result, our ground truth now consists of five answers per question, thus overcoming the limitation of having only one correct answer.
In total, we started with 1087 question-answer pairs. After identifying four similar answers for each correct answer, we had 4264 rows requiring annotation. Following careful annotation, we obtained 2503 pairs labeled as 1 (indicating similarity) and 1761 pairs labeled as 0 (indicating no relevance or similarity). Including the correct answers (1087 rows), our qrel file now comprises 3579 rows, all with relevance label 1, indicating question-answer relevance. The maximum number of relevant answers for a question being 5 and the minimum being 1.

[4]m3hrdadfi/bert-fa-base-uncased-farstail-mean-tokens

Chapter 4

Methods

In this chapter, we will describe the methodology employed in our study. Given our primary focus on the question-answering task, our main model of choice is BERT. As a baseline, we have selected TF-IDF. In this chapter, we will explain our implementation and methodology step by step.

4.1 Ranking models

4.1.1 TF-IDF

For baseline models, since the raw count of a word is not the right metric to be used for Term frequency, we used TF-IDF. Term Frequency-inverse Document Frequency (TF-IDF) is a statistical measure that evaluates the relevance of a word in a document in a collection of documents and eliminate words with a low TF-IDF score.

$$TF - IDF = TF * log(\frac{N}{d_f}) \tag{4.1}$$

- Term Frequency (tf): the frequency of a word in a document. We often use the log representation of the term count as $TF = 1 + log_{10}t_c$
- Inverse Document Frequency (idf): which is defined as $log(\frac{N}{d_f})$ and measures how common or rare a word is in the entire document set.

Multiplying two metrics, TF-IDF measures how relevant a term is to a document in a collection of documents. If the term is common in these documents but is rare in other documents, then the TF-IDF score will be high. Documents with higher TF-IDF scores are considered very relevant to the search term [62]. The *sklearn* package provides a function (TfdfVectorizer) to compute the TF-IDF value for the sentences.
TF-IDF is a conventional method used for keyword weighting, while another algorithm known as Best Match 25 (BM25) offers various variations. BM25 improves upon TF-IDF by casting relevance as a probability problem. According to probabilistic information retrieval, we can define that a relevance score should reflect the probability which satisfied a user needs, and users consider it as a result relevant.

4.1.2 BM25

BM25 is a bag-of-words retrieval function. It scores each document in a collection of documents according to the document's relevance to a term query. A query Q consists of terms $q_1, \ldots, q_n$, so the BM25 score for document D is:

$$BM25(D,Q) = \sum_{i=1}^{n} IDF(q_i, D) + \frac{f(q_i, D)(k_1 + 1)}{f(q_i) + k_1(1 - b + b\frac{|D|}{d_{avg}})} \tag{4.2}$$

Where:

- $f(q_i, D)$ is the number of items term q_i are in document D.
- $|D|$ is the number of words in document D.
- d_{avg} is the average number of words per document.
- b and k_1 are hyperparameters for BM25.

$f(q_i, D)$ illustrates the more times the query term appears in the document, the documents will get a higher score. The parameter k_1 is a hyperparameter, which determines the term frequency saturation characteristic. The higher value for k_1 means that the score for each term can continue to go up by relatively more for more instances of that term. When k_1 is 0, we just calculate the $IDF(q_i)$. By default, k_1 has a value of 1.2.
$|D|/d_{avg}$ in the denominator means that a document which is longer than the average document will result in a bigger denominator, gets a lower score. Then, we have more terms in the document that do not match our input query, the documents will get a lower score. Another hyperparameter, b between 0.0 and 1.0, in the denominator, which is multiplied by the ratio of the document length. Having a bigger b, the effects of the document length compared to the average length are more amplified. To nullify this effect we set b to 0. As for the inverse document frequency part, $IDF(q_i, D)$. For a collection with N documents, inverse document frequency for term q_i is computed as:

$$IDF(q_i, D) = log\frac{N}{df_i} \tag{4.3}$$

To be sure that rarer words will have a higher score and contribute more to the final score, the inverse document frequency part is used, which is similar to that of TF-IDF [63].

4.1.3 BERT - ParsBERT

BERT serves as the second baseline for this study. Since the dataset is in Persian, the well-known Persian BERT model is ParsBERT, which has been trained on a large collection of Persian corpora.
For the first part of this study, we used the "SentenceTransformers" package in Python, which employs dense vector representations. These vector representations, often referred to as embeddings, capture the semantic meaning of the text. The primary concept behind SentenceTransformers is to leverage pre-trained transformer models, such as BERT, to generate high-quality sentence embeddings. These embeddings can be employed for comparing and measuring the similarity between sentences or documents. To measure this similarity, cosine

similarity is utilized [28]. As we will detail in Section 4.1.4 (Question-Question similarity), we employed three distinct BERT models for the Persian language. All of these models are based on ParsBERT but trained on different corpora.
In the second part of our study, we apply question-answering retrieval techniques. we used the "SentenceTransformers" package in Python again, but as will explain in Section 4.1.5 (Question-Answer retrieval), we experimented with two pre-trained QA models for question-answering retrieval: *multi-qa-mpnet-base-dot-v1*[1] and *bert-fa-QA-v1*[2]. By generating high-quality sentence embeddings for questions and answers, we retrieve the relevance between questions and answers by computing cosine similarity. To choose one of these models, after comparing the results, we found that the latter outperformed the former. Therefore, for our experiment, we used 'bert-fa-QA-v1', a pre-trained Persian model specifically designed for question-answering tasks. This model is a fine-tuned version of ParsBERT on the PersianQA dataset.

4.1.4 Question-question similarity

"Question-Question similarity" refers to the measurement or assessment of how similar or related two different questions are to each other. This similarity can be determined through various techniques in natural language processing and information retrieval. It is often used in tasks such as question-answering and information retrieval to identify similar or related questions and provide relevant answers. One of the most commonly used metrics to measure this similarity is cosine similarity.
In our study, to measure the similarity between questions in the dataset, the dataset was split into two parts: 80 percent and 20 percent, based on the timestamps of the questions. We used the 20 percent portion as the query set and identified five similar questions from the remaining 80 percent portion. Both methods require distinct preprocessing steps. For TF-IDF, the preprocessing involves removing Persian stopwords, normalizing, and tokenizing the text, which can be done by importing and using the HAZM library, including functions like *sent_tokenize, word_tokenize, WordEmbedding, and stopwords_list*. In contrast, BERT uses its own internal processing techniques, which make it less reliant on manual preprocessing steps. After preprocessing the data, we employed the TF-IDF technique from the sklearn library. Additionally, we used ParsBERT[3] as well as three other Persian BERT models[4], all of which are based on ParsBERT but were trained on distinct datasets. We computed the top five similar questions for each of these models and then manually annotated their relevance. The process of manual annotation is explained in Section 3.4.1.

4.1.5 Question-answer retrieval

"Question-answer retrieval" is a process in information retrieval and natural language processing (NLP) that involves finding relevant answers to a given question from a collection of documents, databases, or a corpus of text. This process is often associated with question-answering systems and search engines.

[1] https://www.sbert.net
[2] https://huggingface.co/ForutanRad/bert-fa-QA-v1
[3] HooshvareLab/bert-base-parsbert-uncased
[4] m3hrdadfi/bert-fa-base-uncased-wikitriplet-mean-tokens, m3hrdadfi/bert-fa-base-uncased-wikinli-mean-tokens, m3hrdadfi/bert-fa-base-uncased-farstail-mean-tokens

In this phase of the study, our focus was primarily on retrieving the answers for each question. To accomplish this, we employed three methods: TF-IDF, BM25 and BERT[5] for question answering. TF-IDF and BERT were used to compute the cosine similarity between questions and their corresponding answers. BM25 was used for ranking to retrieve answers for each question

In the first step, we retrieve five answers for each question (The Strict evaluation). Since there is only one correct answer for each question, the MAP and MRR computed for the model were very low (Table 5.2, Table 5.3, Table 5.4). To address this issue, we made the assumption that, for each answer, there are similar answers in the collection that can be considered as the true answer for each question (The Lenient evaluation). Therefore, after ranking the question answers, we measured the similarity between each answer, manually annotated them, and added these similar answers to the ground truth (qrel file). Subsequently, we re-measured the MAP and MRR to assess the model's performance with this updated ground truth.

Given the poor performance of the three models, especially with BERT, we have implemented other approach to address this issue. In this approach, we are focusing on measuring recall at four specific points: k = 5, 50, 100 and 200. To accomplish this, we've expanded the scope of our answer retrieval process for each question. We now retrieve a total of 200 answers for each question. Subsequently, we assess recall at the designated points (k = 5, 50, 100 and 200), and also recompute both MAP and MRR to provide a comprehensive evaluation.

4.2 Re-ranking

The dataset is for medical question answering, so it contains numerous medical terms, which can assist models in identifying relevant answers for each question based on these medical terms. However, since we did not have labeled data for these entities in Persian medical datasets, we needed to use methods to automatically annotate them first, then identify new entities, and finally use this labeled data to enhance ranking. With these assumptions, we believe that the occurrence of these medical terms in sentences (questions and answers) is one of the reasonable ways to improve the ranking of models.

4.2.1 Identifying new entities

We applied NER, commonly implemented as token classification [64], to our dataset to determine IOB labels. Given that our dataset is medical question answering, our primary goal is to identify drug and disease names. So, we focus on the names of diseases and drugs, attempting to label these entities.

To achieve this, we extracted the Persian names of all drugs and diseases from Persian Wikipedia and organized them in a pandas DataFrame. Based on these names, we employed the Hazm library to extract drug and disease names through *weak supervision*, using exact matching for extraction. Given the presence of misspellings and typos in the dataset for drugs and diseases, we opted to employ weak supervision methods for labeling. This means that if the exact name of a drug or disease is present in dataset, we extracted it and added it to a list.

Recognizing the potential for misspellings and typos in the dataset, we use entities from Wikipedia as a form of distant supervision. Additionally, we employ and train the ParsBERT model to identify new entities within the dataset. To do this, we partitioned the dataset into

[5] the ForutanRad/bert-fa-QA-v1

two parts: the part with IOB labeling served as the train set, and the other part without IOB labels served as the test set. Using this approach, we are able to predict new entities in the dataset.
Then we need to add these new entities to our dataset. After employing distant supervision labeling to identify new entities in the dataset, upon closer examination, we discovered that some of them were represented by [UNK] tokens when predicting their entities and IOB labels. Therefore, we decided to remove these tokens. However, for other tokens, although they may not represent meaningful words individually, they were part of larger words and can be useful for determining the IOB labels. Therefore, we chose not to clean these new entities and decided to retain their tokens and labels. This approach ensured that even though certain tokens lacked explicit meaning, they still contributed to accurately assigning IOB labels. Incorporating these new entities and their corresponding IOB labels into the dataset is expected to enhance the ranking of retrieval answers for each question. Therefore, the next step involved merging the tokens and their IOB labels to recreate the whole sentence. We combined these fields for both questions and answers and merged them into the main dataset. This merging makes sure that the dataset has the added information, which helps analyze it better and improves how well it works for finding and retrieving answers. Now our dataset consists of the question, answers and its IOB labels.

4.2.2 Improving the ranking with NER labels

Now we need to do re-ranking models and retrieve answers for each question in our dataset using the newly added entities. First, for each question, we computed the cosine similarity between the question and each answer. Then, we counted the IOB labels (B-Drug, I-Drug, B-Disease, and I-Disease) for each entity occurred in the answers. We multiplied the cosine score by one plus the sum of the IOB label numbers occurring in the answer (Cosine Score * (1 + Sum of count of IOB label)). Finally, we retrieved the top k (k = 5, 50, 100, 200) answer for each question based on this combined score, re-ranking the answers for each question.
Based on our previous experiments, we found that TF-IDF achieved the best results in our baseline. Therefore, we applied the TF-IDF re-ranking method to the entire dataset to retrieve answers for each question. Since we are curious about our worst-performing model, we applied the same scenario for ParsBERT too.
The motivation behind implementing this straightforward and simple re-ranking method is due to the sparse nature of the data and the limited number of entities present in both questions and answers. Due to this sparsity, we lack sufficient precise information to perform more complex re-ranking function. As a result, we believe that considering the occurrence of medical entities as a singular measure of relevance could serve as a valuable metric. By incorporating both cosine similarity and the count of IOB labels for entities, we aim to leverage any available tokens for relevance assessment, then it could enhance the retrieval process.

4.3 Metrics

In the realm of information retrieval and classification algorithms, evaluating the performance of models is essential to understanding their effectiveness in identifying and retrieving relevant items. Several key metrics play a crucial role in this evaluation process, each offering unique insights into different aspects of a model's performance. Thus, to evaluate and compare all aspects of models, we typically use a combination of different metrics.

In this section, we explore and define precision, recall, F1-score, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), normalized Discounted Cumulative Gain (nDCG), and Recall@k, which are the most crucial metrics in IR and NLP systems.

Precision, Recall and F1-score:

Precision is a crucial metric that measures the accuracy of positive predictions made by a system. It is calculated as the ratio of true positives to the sum of true positives and false positives. As we can see in Equation 4.4, a high precision score shows that the model is effective at minimizing false positives, making it particularly relevant in scenarios where misclassifying items as positive carries significant consequences [65].

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{4.4}$$

Recall, also referred to as sensitivity, evaluates the model's capability to capture all relevant items. It is computed as the ratio of true positives to the sum of true positives and false negatives. As shown in Equation 4.5, a high recall score indicates that the model is proficient at identifying a significant portion of the relevant items, making it vital in scenarios where the missing relevant items is a significant concern [65].

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{4.5}$$

F1-score is the harmonic mean of precision and recall, providing a balanced measure that considers both false positives and false negatives (Equation 4.6). It is particularly useful when a balance between precision and recall is necessary [65].

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{4.6}$$

Mean Average Precision (MAP):

MAP is a comprehensive metric that takes into account both precision and recall at different positions in a ranked list. It offers a nuanced insight into the effectiveness of a retrieval system in retrieving relevant documents. The MAP score is derived by averaging precision values at various recall levels for each query and then calculating the mean across all queries. In the evaluation of Information Retrieval (IR) models, a higher MAP score shows a more effective system. In simpler words, MAP represents the average precision values obtained for relevant documents at each position in the ranked list, considering both precision and recall [65]. Equation 4.7 illustrates the formula for MAP:

$$MAP = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{\sum_{j=1}^{N_i} \text{precision at } k_j}{\text{Number of relevant Doc in } q_i} \tag{4.7}$$

Where:

- $|Q|$ is the total number of queries in the evaluation dataset.
- q_i is an individual query from the set of queries Q.

- N_i ia the number of relevant documents for the specific query q_i.
- k_j is the position of the j-th relevant document in the ranked list for the query q_i.

MRR (Mean Reciprocal Rank):

MRR (Equation 4.8) emphasizes the importance of retrieving the most relevant document early in the ranked list. It calculates the average of the reciprocal ranks of the first relevant document for each query. MRR is particularly useful when the priority is to provide users with highly relevant information promptly. Similar to MAP, a higher MRR score indicates better performance [65].

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{Rank of the first relevant Doc in } q_i} \tag{4.8}$$

nDCG (Normalized Discounted Cumulative Gain):

nDCG evaluates the quality of a ranking by considering both relevance and the position of relevant documents in the list. It addresses the issue of assigning higher scores to more relevant documents at the top of the list [65].

Recall@k:

Recall@k assesses the proportion of relevant items successfully retrieved within the top k results, providing insights into the model's ability to capture relevant information early in the ranked list [65].

Chapter 5

Results

In this chapter, we present the results of the methods discussed in Chapter 4. The chapter consists of the results for our baseline, which include TF-IDF, BM25, and ParsBERT, to extract similar questions for each question and then retrieve the answer for each question. Additionally, to conduct more experiments on the dataset, we applied group question answer retrieval and question classification. Finally, to extract entities in the dataset, we used Named Entity Recognition (NER), and to re-rank the models, we took into account the weighted entities.

5.1 Question-question retrieval

In Chapter 4, we outlined our methodology for assessing question-question similarity. We calculated MAP and MRR for three models: "TF-IDF", "BERT-based" models (including BERT and another variation trained on a different Persian text collection). Interestingly, as we can see in Table 5.1, despite TF-IDF serving as our baseline model, it outperformed the other BERT variations in terms of both MAP and MRR. When comparing the two Persian BERT variations, ParsBERT demonstrated superior results. Consequently, for the subsequent phase of our study aimed at determining similarity using Persian BERT, we opted to use ParsBERT.

Table 5.1: The results of MAP, nDCG, and MRR for the three models to assess the similarity for each question

Model	**MAP**	**nDCG**	**MRR**
TF-IDF	**0.4629**	**0.5656**	**0.7680**
bert-fa3	0.1882	0.2638	0.5152
parsBERT	0.2598	0.3568	0.5934

5.2 Question–answer retrieval

In Chapter 4, we detailed our methodologies for question-answer retrieval. As evident from the results presented in Table 5.2, when employing the TF-IDF approach for information retrieval tasks with just one correct answer in the qrel file, the MAP and MRR may exhibit similar behavior. One reason for this is that in a system with only one correct answer, the system can achieve perfect precision by returning the single correct answer as the top-ranked result. Then,

both MAP and MRR are influenced by the placement of the correct answer at rank 1. Also, when we have only one correct answer it can simplify the evaluation process. The system is evaluated based on its ability to retrieve and rank a single relevant document accurately. So it can lead to similar MAP and MRR values. However, in more complex scenarios such as having multiple correct answers, the differences between MAP and MRR may become more apparent. For the second part of our experiments, since we explained in Section 3.4.2, we applied TF-IDF to identify similar answers for each correct answer, annotated them carefully and added these comparable answers to the qrel file. We expected that adding more similar answers to the qrel file might lead to an improvement in our results. However, interestingly, adding more similar answers for a correct answer in the qrel file resulted in lower MAP. On the other hand, MRR scores showed a slight improvement compared to the method where only one correct answer was considered strictly.

Since MAP is sensitive to recall at different ranks, when multiple similar correct answers exist, the system needs to consistently retrieve and rank all of them at high positions to achieve a high MAP score. Any miss or lower ranking of relevant documents negatively impacts precision, and reduces the overall MAP score. Similarly, MRR focuses on the rank of the first correct answer, when there are more similar answers, the system has to put all of them at the top. Our results show that this can give the system a higher MRR compared to having only one correct answer.

These results (Table 5.2) for TF-IDF show that adding more similar answers for a correct answer in the qrel file makes the evaluation more stringent. In such cases, the system is assessed not only on its ability to find one correct answer but also on its capability to retrieve and rank all similar correct answers accurately.

Table 5.2: The results of MAP, nDCG, MRR and recall (k= 5, 50, 100, 200) for TF-IDF. The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

	k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Strict	5	0.2077	0.2314	0.2077	0.3035
	50	0.2244	0.2880	0.2244	0.5262
	100	0.2252	0.2980	0.2252	0.5878
	200	0.2257	0.3074	0.2257	0.6559
Lenient	5	0.1466	0.1875	0.2600	0.1957
	50	0.1714	0.2550	0.2780	0.3886
	100	0.1731	0.2693	0.2789	0.4507
	200	0.1741	0.2824	0.2794	0.5160

Table 5.3: The results of MAP, nDCG, MRR and recall (k= 5, 50, 100, 200) for BM25. The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

	k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Strict	5	0.1994	0.2195	0.1994	0.2805
	50	0.2149	0.2735	0.2149	0.4958
	100	0.2159	0.2850	0.2159	0.5666
	200	0.2164	0.2947	0.2164	0.6366
Lenient	5	0.1313	0.1692	0.2432	0.1741
	50	0.1533	0.2319	0.2609	0.3542
	100	0.1552	0.2483	0.2618	0.4243
	200	0.1563	0.2632	0.2624	0.4978

Since we applied BM25 and BERT in our experiment as well, we employed the same scenario for BM25 and BERT. For both of these models, as shown in Table 5.3 and Table 5.4, we observe equal MAP and MRR scores when we have only one correct answer. The results for BM25 are similar to TF-IDF since both of these models use term frequency. But, the results of BERT, when compared to TF-IDF and BM25, are understated. Additionally, we implemented the strategy of adding similar answers to our qrel file. However, the results indicate that the performance does not improve in the case where we have only one correct answer in the qrel file. Comparing TF-IDF and BERT reveals that TF-IDF can yield better results, even when we have only one correct answer in our qrel file.

Table 5.4: The results of MAP, nDCG, MRR and recall (k= 5, 50, 100, 200) for BERT. The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

	k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Strict	5	0.0230	0.0269	0.0230	0.0386
	50	0.0273	0.0444	0.0273	0.1149
	100	0.0280	0.0534	0.0280	0.1711
	200	0.0287	0.0672	0.0287	0.2704
Lenient	5	0.0139	0.0200	0.0316	0.0222
	50	0.0172	0.0364	0.0383	0.0736
	100	0.0180	0.0457	0.0394	0.1158
	200	0.0188	0.0603	0.0403	0.1921

5.3 Grouped question-answer retrieval

As explained in Section 3.1, the dataset has its own categories for questions. In our preprocessing phase (Section 3.2), we merged some identical categories, reducing the total number of categories from 36 to 25. For this part of our experiments, we decided to use the existing categories. We narrowed down the range of possible answers to the respective categories. Therefore, for each question, we filtered the expected answer domain to its corresponding category and then applied the models to retrieve the relevant answer for each question. As

evident from the tables (Table 5.5, Table 5.6, and Table 5.7), the performance of all models improved.
After evaluating the performance of various retrieval models in both grouped and non-grouped settings, Table 5.8 provides a comprehensive comparison of these methods. It illustrates their MAP, nDCG, MRR and Recall@k scores at k = 200, showcasing the best results for all the models in both grouped and non-grouped scenarios, considering both strict and lenient evaluations.
In the non-grouped scenario, the TF-IDF model exhibited strong performance across all metrics in both strict and lenient evaluations. It achieved a MAP of 0.2257 in strict conditions and 0.1741 in lenient conditions. Similarly, BM25 also demonstrated similar results, with MAP values of 0.2164 (strict) and 0.1563 (lenient). However, the BERT model get lower scores for all evaluation metrics. Moreover, in the grouped scenario, TF-IDF outperformed other models, achieving a higher MAP of 0.3430 (strict) and 0.2283 (lenient) again. BM25 also maintained competitive performance with MAP values of 0.3235 (strict) and 0.2115 (lenient). But intresting thing is about BERT which has significant improve compered to the non-grouped scenario, but still lower than TF-IDF and BM25.

Table 5.5: The results of MAP, nDCG, MRR and recall (k = 5, 50, 100, 200) for TFIDF-Grouped. The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

	k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Strict	5	0.3181	0.3493	0.3181	0.4443
	50	0.4091	0.4277	0.4091	0.7543
	100	0.3427	0.4467	0.3427	0.8702
	200	0.3430	0.4532	0.3430	0.9172
Lenient	5	0.1918	0.2494	0.3583	0.2579
	50	0.2247	0.3396	0.3825	0.5126
	100	0.2275	0.3612	0.3838	0.6048
	200	0.2283	0.3729	0.3843	0.6588

Table 5.6: The results of MAP, nDCG, MRR and recall (k = 5, 50, 100, 200) for BM25-Grouped. The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

	k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Strict	5	0.2978	0.3286	0.2978	0.4213
	50	0.3218	0.4142	0.3218	0.7663
	100	0.3230	0.4294	0.3230	0.8620
	200	0.3235	0.4383	0.3235	0.9245
Lenient	5	0.1746	0.2311	0.3388	0.2410
	50	0.2077	0.3253	0.3645	0.5085
	100	0.2104	0.3461	0.3654	0.5971
	200	0.2115	0.3603	0.3660	0.6638

Table 5.7: The results of MAP, nDCG, MRR and recall (k = 5, 50, 100, 200) for BERT-Grouped. The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

	k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Strict	5	0.0988	0.1160	0.0988	0.1683
	50	0.1263	0.2178	0.1263	0.5878
	100	0.1288	0.2471	0.1288	0.7690
	200	0.1294	0.2596	0.1294	0.8592
Lenient	5	0.0588	0.0872	0.1284	0.1024
	50	0.0867	0.1810	0.1575	0.3858
	100	0.0902	0.2109	0.1597	0.5190
	200	0.0912	0.2262	0.1604	0.5950

Table 5.8: Comparison of all models at k = 200 for optimal retrieval in Strict and Lenient evaluations, Across Non-Grouped and Grouped scenarios. Grouped scenario: questions are filtered based on their current category in the dataset.

		Model	**MAP**	**nDCG**	**MRR**	**Recall@k**
Non-Grouped	Strict	TF-IDF	**0.2257**	**0.3074**	**0.2257**	**0.6559**
		BM25	0.2164	0.2947	0.2164	0.6366
		BERT	0.0287	0.0672	0.0287	0.2704
	Lenient	TF-IDF	**0.1741**	**0.2824**	**0.2794**	**0.5160**
		BM25	0.1563	0.2632	0.2624	0.4978
		BERT	0.0188	0.0603	0.0403	0.1921
Grouped	Strict	TF-IDF	**0.3430**	**0.4532**	**0.3430**	**0.9172**
		BM25	0.3235	0.4383	0.3235	0.9245
		BERT	0.1294	0.2596	0.1294	0.8592
	Lenient	TF-IDF	**0.2283**	**0.3729**	**0.3843**	**0.6588**
		BM25	0.2115	0.3603	0.3660	0.6638
		BERT	0.0912	0.2262	0.1604	0.5950

5.4 Question classification

Text classification is another crucial task in the field of NLP. The objective of this task is to label text, including paragraphs and sentences. There is a wide range of applications, including question answering, that leverage text classification [66]. In our dataset, the goal of text classification is to assign labels of group to each question.

Since each question belongs to a particular group, the labels are easily available. For this task, we split the dataset into 80 percent for the training set and 20 percent for the test set.

For the first experiment in this section, we used TF-IDF to extract features. Subsequently, we applied two different classifiers and compared their performance to identify the best classifier. These classifiers include Multinomial Naive Bayes and Linear SVM. As indicated in Table 5.9, the SVM achieved better Recall, Precision, and F1 score.

Then, we fine-tuned the dataset using BERT[1]. After 10 epochs, the results show that BERT achieves a precision of 0.68, recall of 0.69, and an F1-score of 0.67 (Table 5.9). This performance

[1]https://huggingface.co/ForutanRad/bert-fa-QA-v1

Table 5.9: The results of Precision, Recall and F1 Score for two different classifiers with TF-IDF features, and for BERT

Classifier	**Precision**	**Recall**	**F1 Score**
Multinomial Naive Bayes	0.18	0.35	0.19
Linear SVM	0.58	0.64	0.58
BERT	**0.68**	**0.69**	**0.67**

is better than the best result obtained with the TF-IDF and Linear SVM classifier.
By classifying questions, the initial step when users pose new queries is to identify the classes to which these questions belong. Afterward, it is advantageous to search for similar questions within the identified class and retrieve corresponding answers. As demonstrated in the grouped retrieval of questions (Table 5.8), the performance of all models tend to improve significantly when the search is limited to specific groups that align with the categorization of the questions. So, it can improve the performance of question answering systems.

5.5 Named entity recognition

In this part of our experiments, we used Named Entity Recognition (NER) to investigate the dataset, defining new IOB labels according to the specific entities in the dataset. The count of IOB labels for each question and answer is presented in Table 5.10. Also, the examples of entities (drug and disease names) and their translations from Wikipedia for Q/A is illustrated in Table 5.11. We merged the two dataframes for questions and answers into a single dataframe and applied fine-tuning on ParsBERT. To do this, we divided the dataset into training, validation, and test sets and classified the tokens accordingly. Since the dataset is not large enough, we used the train_test_split function from the sklearn library. For the training set, we used 80 percent of the dataset, and for both the test set and validation set, we allocated 10 percent of the dataset to each. Consequently, the size of the training set is 1723 records, the validation set consists of 215 records, and the test set consists of 216 records.

Table 5.10: The statistics of IOB-labeled text using entities from Wikipedia for Q/A

Label	**Question**	**Answer**
B-Disease	205	95
I-Disease	2	0
B-Drug	166	42
I-Drug	137	25
O	70967	13866

Since the accuracy metric can be misleading, particularly when a significant portion of labels are 'outside' (O), it is crucial to examine the precision, recall, and F1-score for individual labels. Additionally, we excluded the 'O' label and evaluated the model specifically for 'I' and 'B' labels. The results are presented in Table 5.12.

Table 5.11: Showing examples of entities (drug and disease names) and and their translations from Wikipedia for Q/A

Drugs-Name	Translation	Disease-Name	Translation
پروژسترون	Progesterone	وبا	Plague,
نورتریپتیلین	Nortriptyline	اسکیزوفرنی	Schizophrenia
کلوتریمازول	Clotrimazole	زگیل	Wart
استامینوفن	Acetaminophen	آبسه	Abscess
آزیترومایسین	Azithromycin	سرطان	Cancer
جینسینگ	Ginseng	ایدز	AIDS
متفورمین	Metformin	افسردگی	Depression
دیازپام	Diazepam	بواسیر	Hemorrhoids
ریسپریدون	Risperidone,	فتق	Hernia
پروژسترون	Progesterone	تبخال	Herpes
فلوکستین	Fluoxetine	کرونا	Coronavirus
لووتیروکسین	Levothyroxine	هپاتیت	Hepatitis
کلوتریمازول	Clotrimazole	پولیپ	Polyp
پروپرانولول	Propranolol	اسهال	Diarrhea
فلووکسامین	Fluvoxamine	واریس	Varis

Table 5.12: The results of Precision, Recall and F1 score of NER

Label	**precision**	**Recall**	**F1 Score**	**support**
B-Drug	1.00	1.00	1.00	21
B-Disease	0.95	0.74	0.83	27
I-Drug	1.00	0.52	0.68	81
I-Disease	0.00	0.00	0.00	0
macro avg	0.74	0.56	0.63	129
weighted avg	0.99	0.64	0.77	129

The model did not yield satisfactory results after the first epoch, but a notable improvement was observed after increasing the number of epochs to 20.
An interesting aspect of this dataset is that, due to the limited number of certain labels such as 'I-Disease' in questions and 'I-Disease', 'I-Drug', and 'B-Drug' in answers, particularly for 'I-Disease', the model struggles to train effectively on these labels. This issue arises when the dataset is imbalanced. As shown in Table 5.10, the dataset is imbalanced for these labels. The number of 'B-Disease' labels is higher than others in both questions and answers. Another potential reason for this challenge could be the insufficient training data for some labels, stemming from the imbalanced labeling in the dataset, which makes it difficult for the model to learn how to identify them correctly.

5.5.1 Identifying new entities

For the next step, we filter the dataset for sentences that have 'I' and 'B' labels. We use this subset of data as the training set and proceed to train the model on it. To identify new entities, we apply the trained model to the remaining data, which serves as the test set. After training the model for 5 epochs, it identified 278 new entities in the test set. Among these new entities, 223 are unique. So, for the set of questions, there are a total of 131 new entities that 109 are unique. And in the set of answers, there are 147 new entities, with 126 being unique. The statistical results for new entities with their IOB labels, are presented in Table 5.13. Also, Table 5.14 shows the newly identified entities and their translation in the dataset.
Before doing the statistics, we examined the new entities and their IOB labels. We observed the presence of tokens labeled as [UNK], indicating unknown tokens in our results. It appears that

Table 5.13: The statistics of new entities

Label	**New entities (Question)**	**New entities (Answer)**	**Unique new entities (Question)**	**Unique new entities (Answers)**
B-Disease	20	40	17	30
B-Drug	15	14	14	14
I-Disease	19	28	15	20
I-Drug	77	65	63	62

Table 5.14: Showing examples of new entities, IOB labels ad their translation

IOB Label	New_Entity	Translation
B-Disease	hiv	HIV
I-Disease	hpv	HPV
B-Disease	استروژن	Estrogen
I-Disease	نازایی	Infertility
B-Disease	مسمومیت	Poisoning
I-Disease	الرژی	Allergy
I-Drug	ید	Iodine
B-Disease	فشارخون	Blood pressure
B-Disease	دیسک	Disc
B-Drug	زعفران	Saffron
B-Disease	قندخون	Blood sugar
B-Drug	بکارت	Virginity
I-Disease	اعتیاد	Addiction
B-Disease	قارچ	Fungus disease

the BERT model we used might have a fixed vocabulary dictionary instead of using subtokens for tokenization. This limitation results in the presence of [UNK] tokens, necessitating their removal. By looking closer to the remaining tokens and comparing them with sentences in our dataset, we determined that removing certain tokens is not a viable strategy for cleaning the new entities. Consequently, we opted to retain them and incorporated these new entities and IOB labels into our dataset, which already contained IOB labels from weak supervision. The weak supervision method successfully extracted the names of drugs and diseases from the dataset. As a result, we now have additional labels for some entities, although they may not be completely accurate and exact words.

5.5.2 Improving the ranking with NER labels

In the previous section, we evaluated NER, found new entities in our dataset and converted the entities to IOB labeling. Now, we will use this information to improve our rankers. To improve the model, as mentioned in the previous section, we added new entities and their IOB labels to the dataset. The baselines we applied in the first steps of our experiments were TF-IDF, BERT, and BM25 to retrieve the answer for each question. Since TF-IDF yielded the best results among our baselines, we combined the NER re-ranking with the TF-IDF model.
As we can see in Table 5.15, the result of the re-ranked TF-IDF model does not show any significant changes. Also, we conducted the same scenario for the grouped Question-Answering retrieval, and the results remained unchanged. Therefore, we did not include the results in the table.

There are several reasons why IOB entities may not have a significant impact on the TF-IDF model:

- Insufficient presence of IOB entities: It's possible that our dataset lacks a good presence of IOB entities that effectively bridge the gap between questions and their corresponding answers during retrieval. Also, TF-IDF's strength in identifying specific terms may overshadow the influence of IOB entities, particularly if these entities are not strongly associated with the terms used for retrieval.
- Limited influence of IOB entities: Sometimes the incorporation of IOB entity counts as an additional feature may not substantially modify the similarity scores enough to alter the ranking of documents significantly. It can be because TF-IDF vectors might already encapsulate the majority of relevant information necessary for retrieval, thereby disregarding the impact of IOB entities.
- Simple choice of re-ranking function: As we can see in our re ranking Section 4.2, we used a simple function for counting entities and re-ranking (Cosine Score * (1 + Sum of count of IOB label).

Table 5.15: The results of MAP, nDCG, MRR and recall (k= 5, 50, 100, 200) for TF-IDF re-ranked (Adding the number of medical entities to the score). The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

		k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Baseline	Strict	5	0.2077	0.2314	0.2077	0.3035
		50	0.2244	0.2880	0.2244	0.5262
		100	0.2252	0.2980	0.2252	0.5878
		200	0.2257	0.3074	0.2257	0.6559
	Lenient	5	0.1466	0.1875	0.2600	0.1957
		50	0.1714	0.2550	0.2780	0.3886
		100	0.1731	0.2693	0.2789	0.4507
		200	0.1741	0.2824	0.2794	0.5160
Grouped	Strict	5	0.3181	0.3493	0.3181	0.4443
		50	0.4091	0.4277	0.4091	0.7543
		100	0.3427	0.4467	0.3427	0.8702
		200	0.3430	0.4532	0.3430	0.9172
	Lenient	5	0.1918	0.2494	0.3583	0.2579
		50	0.2247	0.3396	0.3825	0.5126
		100	0.2275	0.3612	0.3838	0.6048
		200	0.2283	0.3729	0.3843	0.6588
Re-ranked	Strict	5	0.2042	0.2288	0.2042	0.3035
		50	0.2202	0.2847	0.2202	0.5262
		100	0.2211	0.2944	0.2211	0.5869
		200	0.2215	0.3035	0.2215	0.6522
	Lenient	5	0.1449	0.1860	0.2562	0.1960
		50	0.1686	0.2524	0.2734	0.3884
		100	0.1704	0.2666	0.2744	0.4499
		200	0.1713	0.2796	0.2749	0.5140

To gain a different perspective on the impact of IOB entities in our dataset, we were curious to assess their effect on our weakest model, ParsBERT. We applied the same strategy to BERT, and interestingly observed that adding the number of IOB labels occurring in the answers to the score had a positive impact on our BERT model. As illustrated in Table 5.16, including these entities improved the results for the BERT model. The improvement observed in the performance of the BERT model after adding the number of IOB entities from the answer can be due to some reasons:

- BERT is a contextualized language model that gets rich semantic relationships in text. By considering the number of IOB entities in the answer, BERT gains additional contextual information about the content, which helps it better understand the relevance of the answer to the given question.
- Including the count of IOB entities in the answer as a feature can improve the relevance signal for BERT. This feature provides BERT with more information to distinguish between relevant and irrelevant answers, which leads to improved retrieval performance.

Table 5.16: The results of MAP, nDCG, MRR and recall (k= 5, 50, 100, 200) for ParsBERT re-ranked (Adding the number of medical entities to the score). The strict evaluation setting is the setting with only one relevant answer per question (the original answer from the doctor). The lenient evaluation setting is the setting with at most 5 expanded answers per relevant answer

		k	**MAP**	**nDCG**	**MRR**	**Recall@k**
Baseline	Strict	5	0.0230	0.0269	0.0230	0.0386
		50	0.0273	0.0444	0.0273	0.1149
		100	0.0280	0.0534	0.0280	0.1711
		200	0.0287	0.0672	0.0287	0.2704
	Lenient	5	0.0139	0.0200	0.0316	0.0222
		50	0.0172	0.0364	0.0383	0.0736
		100	0.0180	0.0457	0.0394	0.1158
		200	0.0188	0.0603	0.0403	0.1921
Re-ranked	Strict	5	0.0518	0.0573	0.0518	0.0745
		50	0.0587	0.0847	0.0587	0.1913
		100	0.0596	0.0956	0.0596	0.2594
		200	0.0604	0.1107	0.0604	0.3670
	Lenient	5	0.0281	0.0402	0.0644	0.0425
		50	0.0340	0.0657	0.0746	0.1219
		100	0.0352	0.0785	0.0760	0.1796
		200	0.0362	0.0957	0.0769	0.2670

Finally, adding the number of IOB entities to the grouped models does not have any impact on the models. This may be due to the limitation of the answers in each category, which does not significantly influence the results of retrieving answers.

Chapter 6

Discussion

In this chapter, we will explore a detailed analysis using the experiment outcomes discussed in Chapter 5. Our discussion will encompass the challenges of data, the retrieval models, and the impact of different models on our dataset. We will also explore the effects of strict and lenient relevance labels, as well as the impact of grouping data on our dataset. Finally, we will consider NER's impact on re-ranking the model.

6.1 Analysing the challenges

Data

As we thoroughly explained in Chapter 3 about the challenges we encountered with the data, we will briefly highlight some of them in this section as well.
Given that the data originates from a Persian medical question-answering context, it is apparent that the data is not entirely clean. In addition to common Persian stopwords, it contains numerous unwanted words and phrases, such as "Hi", "How are you", "I have a question", "Thank you" and others of a similar nature. Therefore, the initial challenge revolves around determining which words and phrases to remove from the dataset. Another challenge was the number of categories. As discussed in Section 3.1, our dataset contains 36 categories. To address the imbalance in question distribution across these categories, we opted to merge certain categories with less data into others that were more abundant and highly similar. Another interesting issue about this dataset is the occurrence of long questions with short, uninformative answers. Sometimes these answers are as simple as "yes" or "no."
However, when it comes to assessing the similarity between questions or extracting answers from them, we encounter certain challenges that we explain bellow:

Question-question similarity

- In typical question answering systems, the length of the question is usually shorter than the length of the answer. However, interestingly, in this dataset, there is an issue regarding very lengthy questions. In such cases, both models (TF-IDF and BERT) often retrieve similar questions or retrieve answers that are not necessarily similar or relevant. This occurs because lengthy questions may contain numerous terms and discussions on various topics, including aspects that may not be directly related to the primary concern.

- Some questions are exceptionally unique, making it challenging for both models, TF-IDF and BERT, to find exact or even similar question. In such cases, the models struggle to provide relevant responses due to the rarity and distinctiveness of the questions. So, the presence of unique and uncommon questions that do not have similar counterparts in the dataset can lead to lower MAP and MRR scores for both TF-IDF and BERT models in a question-answering system. This highlights the significance of having a robust and diverse dataset that includes a broad range of questions, including uncommon or unique ones, to enhance the overall performance of the question-answering system.

Question-answer retrieval

In this study, we found that BERT performed better in the classification task, while TF-IDF outperformed BERT in the other tasks. So, in question-answering tasks, TF-IDF achieved better results than BERT. This improved performance of TF-IDF could be attributed to specific terms that appear frequently in medical texts. In some questions containing these specialized terms, TF-IDF may outperform BERT in retrieving more accurate answers. Other research has also shown the limited capabilities of zero-shot use of BERT rankers. Only with sufficient training data they can outperform a lexical baseline. A similar experience, as demonstrated in the legal domain [67], shows that TF-IDF can outperform other methods [68]. Also, the reason why BERT could not perform well in this task is because the embedding representations of ParsBERT have not been trained on the medical domain. As a result, the model might not have good representations of medical terms.
We faced with low MAP and MRR in our first approach for both models. We anticipated that by adding similar answers to the qrel file, we would observe improvements for the models. However, interestingly, not only did we fail to see improvement, but the results also slightly declined.
Our challenges are:

- To perform lenient evaluation, there are certain unique answers for which the model couldn't find similarities to add to the qrel file. As a result, the number of correct answers for these answers remains at one in the qrel file.

- For TF-IDF, we observed improved similarity scores for both question-question and answer-answer pairs when sentences contained similar or specific terms.

- Regarding BERT, since it is used for Persian language, we employed ParsBERT. For some sentence pairs in question-question and answer-answer similarity, despite high scores, the sentences did not exhibit strong similarity. In some cases, sentences were completely dissimilar.

- For question-answer pairs, the challenge arises because the terms appearing in the question do not necessarily appear in the answer. Consequently, TF-IDF does not perform well in such cases.

- Despite experimenting with two BERT models, one for multilingual QA and one for Persian QA, we found that the results and similarity scores were unsatisfactory

Grouped question-answer retrieval

Since the dataset has its own categories, the challenges we faced with Grouped question-answering include:

- There are instances where the question is not in a correct category, resulting in answers such as "it is not my expertise, you need to ask your question to another specialist." Moreover, there are answers instructing the individual to make an appointment to visit the doctor, which adds complexity to the dataset.
- In some cases, questions are asked in two or more categories, leading to answers that may not be very informative or are answered by two different specialists.

Question-answer retrieval using NER

- Using Named Entity Recognition and IOB labels could potentially assist the model in finding more relevant answers for each question. However, this approach did not significantly impact the quality of the models, particularly for TF-IDF. It appears that for TF-IDF, the model, which is based on term frequency, could identify specific terms without relying on IOB labels. But, for ParsBERT, a model that was not pre-trained on medical collections, using medical entities and IOB labels could potentially improve its performance.
- Another issue with ParsBERT is that it appears the model we used may have a fixed vocabulary dictionary instead of using subtokens for tokenization. Consequently, this issue results in the presence of [UNK] tokens, which we had to remove.
- Since the dataset has its own categories, using these categories to find relevant answers for each question posed a challenge. Due to the low number of questions in some categories, retrieving five or more similar answers for each question, or retrieving more than five answers for each question in its category, was difficult. Additionally, in some categories, there may not be IOB labels available, and weighting these entities may not have influenced the results of the grouped models.

6.2 Limitations

In this thesis, we encountered some limitations:

- The dataset we used did not contain a large volume of data, which made training models like ParsBERT more challenging. Thus, the models did not have access to sufficient data, resulting in insufficient performance, especially for models like BERT.
- Another limitation of this thesis is the absence of a Persian BERT model pre-trained on medical datasets. While there was a model named SINA BERT available, we were unable to locate any code or model in platforms like HuggingFace or elsewhere.
- Since this task relies on labeled data, the absence of labeled data for Persian medical NER presents a significant challenge, and makes the task complicated.

Chapter 7

Conclusion

In this thesis, we conducted research on retrieving answers in a Persian medical question answering dataset. The dataset consists of questions posed by ordinary people and answers provided by specialists and doctors. We had several challenges with the dataset, such as lengthy questions compared to very short and unclear answers. Initially, we used well-known similarity models such as TF-IDF and ParsBERT to identify similar questions. Subsequently, we annotated these similar questions and evaluated our models. Similarly, for expanding the answers, we followed the same procedure, identifying five similar answers and annotating them to establish the ground truth for correct answers. We then employed TF-IDF, BM25, and ParsBERT to retrieve the answers. Interestingly, we discovered that although TF-IDF is a basic method for answer retrieval, it got better results compared to ParsBERT. As the dataset is categorized, we applied these retrieval models accordingly and found that TF-IDF outperformed the others. Finally, we extracted drug and disease entities from the dataset and labeled them using IOB labeling. We then applied this labeled data to the best-performing model (TF-IDF) and weighted our entities based on the number of their occurrence in the answers. Surprisingly, the results for our best model (TF-IDF) remained unchanged. However, we were curious about the impact of this strategy on the worst-performing model (ParsBERT) and found that applying this approach had a positive impact on the results of ParsBERT.

7.1 Answers to research questions

Question I: How effective are various methods in identifying similar questions within our dataset?
Since the dataset was challenging and was about medical question answering, we expected the BERT-based models to get better results. But interestingly, TF-IDF outperformed other models. This could be due to the presence of specific terms in the dataset, which TF-IDF could identify more effectively. Thus, term-based retrieval models, which rely on matching query terms with terms in the documents to retrieve similar results, achieved better performance.
Question II: To what extent are various methods effective in retrieving the relevant answer for each question?
Due to the complexity of the dataset and the presence of vague answers for some questions, retrieving relevant answers for each question had significant challenges. However, as shown in our first research question, term-based models got better results compared to BERT-based models like ParsBERT. It appears that the model we selected (ParsBERT) was not trained on medical collections, which exacerbated the difficulty of the task.

Question III: To what extent can question classification into topical categories help the retrieval tasks?
Since the dataset has its own categories, using these categories helped us improve the performance of our models. This could be due to the increased relevance of answers to each question within their respective categories, which helps enable the models to search and retrieve answers more efficiently. This improvement was observed for all our models. Particularly for ParsBERT. Although the improvement was not significant for ParsBERT, grouping the data helped ParsBERT achieve four times better than when it was not grouped. This shows that categorizing the data helps the retrieval of answers, particularly for models like BERT. Therefore, classifying the data can help identify the relevant category for each question and improve the results of answer retrieval.
Question IV: How can we enhance the performance of each model through reranking using medical terms?
Implementing a strategy of identifying entities (IOB labels) within a dataset and assigning weights to medical them to find relevant answers for each question could improve the results. We applied this strategy, and interestingly, it did not have a significant impact on term-based models such as TF-IDF. This might be because TF-IDF can already find specific terms in questions and answers and weighting them accordingly. So, these entities and IOB labels may have already been weighted in TF-IDF. However, for the poorer-performing model, BERT, weighting these entities was beneficial. This shows that since ParsBERT was not pre-trained on medical collections, implementing this strategy helps it to better identify medical entities and the results improved about two times.

7.2 Future work

Based on the implemented methods and using various models within this dataset, and considering the limitations we faced during this experiment, there are several directions for future work:

- Establishing connections between disease and its corresponding drug name within the dataset can significantly enhance the effectiveness of question answering systems. So the system can prioritize answers that address these specific relationships.
- One of the limitations in our research was the scarcity of data. Therefore, using a large and diverse dataset containing various questions and their corresponding answers could be beneficial. The presence of unique questions and answers often presents challenges for the model in identifying similar or relevant answers. So this leads to lower performance results. Using a wide range of datasets can resolve this issue.
- Using a large collection of medical datasets and pre-training ParsBERT specifically on this data could help medical question answering, particularly for datasets in languages other than English, such as Persian.

References

[1] F. Kunneman, T. C. Ferreira, E. Krahmer, and A. Van Den Bosch, "Question similarity in community question answering: A systematic exploration of preprocessing methods and models," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pp. 593–601, 2019.

[2] H. Veisi and H. F. Shandi, "A persian medical question answering system," *International Journal on Artificial Intelligence Tools*, vol. 29, no. 06, p. 2050019, 2020.

[3] Q. Jin, Z. Yuan, G. Xiong, Q. Yu, H. Ying, C. Tan, M. Chen, S. Huang, X. Liu, and S. Yu, "Biomedical question answering: a survey of approaches and challenges," *ACM Computing Surveys (CSUR)*, vol. 55, no. 2, pp. 1–36, 2022.

[4] N. Abadani, J. Mozafari, A. Fatemi, M. Nematbakhsh, and A. Kazemi, "Parsquad: Persian question answering dataset based on machine translation of squad 2.0," *International Journal of Web Research*, vol. 4, no. 1, pp. 34–46, 2021.

[5] M. Farahani, M. Gharachorloo, M. Farahani, and M. Manthouri, "Parsbert: Transformer-based model for persian language understanding," *Neural Processing Letters*, vol. 53, pp. 3831–3847, 2021.

[6] N. Ansari and R. Sharma, "Identifying semantically duplicate questions using data science approach: A quora case study," *arXiv preprint arXiv:2004.11694*, 2020.

[7] M.-D. Olvera-Lobo and J. Gutiérrez-Artacho, "Question answering track evaluation in trec, clef and ntcir," in *New Contributions in Information Systems and Technologies: Volume 1*, pp. 13–22, Springer, 2015.

[8] E. Mutabazi, J. Ni, G. Tang, and W. Cao, "A review on medical textual question answering systems based on deep learning approaches," *Applied Sciences*, vol. 11, no. 12, p. 5456, 2021.

[9] M. Yu, W. Yin, K. S. Hasan, C. dos Santos, B. Xiang, and B. Zhou, "Improved neural relation detection for knowledge base question answering," in *55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*, pp. 571–581, Association for Computational Linguistics (ACL), 2017.

[10] Z. Abbasiantaeb and S. Momtazi, "Text-based question answering from information retrieval and deep neural network perspectives: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, no. 6, p. e1412, 2021.

[11] R. Barskar, G. F. Ahmed, and N. Barskar, "An approach for extracting exact answers to question answering (qa) system for english sentences," *Procedia Engineering*, vol. 30, pp. 1187–1194, 2012.

[12] M. Gupta, N. Kulkarni, R. Chanda, A. Rayasam, and Z. C. Lipton, "Amazonqa: A review-based question answering task," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 4996–5002, International Joint Conferences on Artificial Intelligence Organization, 7 2019.

[13] V. Lopez, V. Uren, E. Motta, and M. Pasin, "Aqualog: An ontology-driven question answering system for organizational semantic intranets," *Journal of Web Semantics*, vol. 5, no. 2, pp. 72–105, 2007.

[14] T. Hamon, N. Grabar, and F. Mougin, "Natural language question analysis for querying biomedical linked data," *Natural Language Question Analysis for Querying Biomedical Linked Data*, 2014.

[15] J. Piskorski and R. Yangarber, "Information extraction: Past, present and future," *Multi-source, multilingual information extraction and summarization*, pp. 23–49, 2013.

[16] Y. Sankarasubramaniam, K. Ramanathan, and S. Ghosh, "Text summarization using wikipedia," *Information Processing & Management*, vol. 50, no. 3, pp. 443–461, 2014.

[17] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 2, pp. 604–624, 2020.

[18] M. A. C. Soares and F. S. Parreiras, "A literature review on question answering techniques, paradigms and systems," *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 6, pp. 635–646, 2020.

[19] S. Badugu and R. Manivannan, "A study on different closed domain question answering approaches," *International Journal of Speech Technology*, vol. 23, no. 2, pp. 315–325, 2020.

[20] S. Mittal and A. Mittal, "Versatile question answering systems: seeing in synthesis," *International journal of intelligent information and database systems*, vol. 5, no. 2, pp. 119–142, 2011.

[21] S. W.-t. Yih, M.-W. Chang, C. Meek, and A. Pastusiak, "Question answering using enhanced lexical semantic models," in *Proceedings of the 51st Annual Meeting of the Association for Computational linguistics*, 2013.

[22] Y. Yang, W.-t. Yih, and C. Meek, "Wikiqa: A challenge dataset for open-domain question answering," in *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 2013–2018, 2015.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (J. Burstein, C. Doran, and

T. Solorio, eds.), (Minneapolis, Minnesota), pp. 4171–4186, Association for Computational Linguistics, June 2019.

[24] P. Sunilkumar and A. P. Shaji, "A survey on semantic similarity," in *2019 International Conference on Advances in Computing, Communication and Control (ICAC3)*, pp. 1–8, IEEE, 2019.

[25] H. C. Wu, R. W. P. Luk, K. F. Wong, and K. L. Kwok, "Interpreting tf-idf term weights as making relevance decisions," *ACM Transactions on Information Systems (TOIS)*, vol. 26, no. 3, pp. 1–37, 2008.

[26] S. M. Mohammad and G. Hirst, "Distributional measures of semantic distance: A survey," *arXiv preprint arXiv:1203.1858*, 2012.

[27] W. Song, M. Feng, N. Gu, and L. Wenyin, "Question similarity calculation for faq answering," in *Third International Conference on Semantics, Knowledge and Grid (SKG 2007)*, pp. 298–301, IEEE, 2007.

[28] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, 2019.

[29] X. Zhang, J. Wu, Z. He, X. Liu, and Y. Su, "Medical exam question answering with large-scale reading comprehension," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[30] Z. Hu, Z. Zhang, H. Yang, Q. Chen, and D. Zuo, "A deep learning approach for predicting the quality of online health expert question-answering services," *Journal of biomedical informatics*, vol. 71, pp. 241–253, 2017.

[31] S. Fox and M. Duggan, "Tracking for health," 2013.

[32] T. R. Goodwin and S. M. Harabagiu, "Medical question answering for clinical decision support," in *Proceedings of the 25th ACM international on conference on information and knowledge management*, pp. 297–306, 2016.

[33] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional attention flow for machine comprehension," in *International Conference on Learning Representations*, 2016.

[34] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "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)*, (New Orleans, Louisiana), pp. 2227–2237, Association for Computational Linguistics, June 2018.

[35] L. Zhou, J. Gao, D. Li, and H.-Y. Shum, "The design and implementation of xiaoice, an empathetic social chatbot," *Computational Linguistics*, vol. 46, no. 1, pp. 53–93, 2020.

[36] Y. Niu, G. Hirst, G. McArthur, and P. Rodriguez-Gianolli, "Answering clinical questions with role identification," in *Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine*, pp. 73–80, 2003.

[37] D. Demner-Fushman and J. Lin, "Knowledge extraction for clinical question answering: Preliminary results," in *Proceedings of the AAAI-05 Workshop on Question Answering in Restricted Domains*, pp. 9–13, AAAI Press (American Association for Artificial Intelligence) Pittsburgh, PA, 2005.

[38] D. Demner-Fushman and J. Lin, "Answering clinical questions with knowledge-based and statistical techniques," *Computational Linguistics*, vol. 33, no. 1, pp. 63–103, 2007.

[39] H. Yu, M. Lee, D. Kaufman, J. Ely, J. A. Osheroff, G. Hripcsak, and J. Cimino, "Development, implementation, and a cognitive evaluation of a definitional question answering system for physicians," *Journal of biomedical informatics*, vol. 40, no. 3, pp. 236–251, 2007.

[40] S. Cruchet, A. Gaudinat, and C. Boyer, "Supervised approach to recognize question type in a qa system for health," *Studies in Health Technology and Informatics*, vol. 136, p. 407, 2008.

[41] Y. Cao, F. Liu, P. Simpson, L. Antieau, A. Bennett, J. J. Cimino, J. Ely, and H. Yu, "Askhermes: An online question answering system for complex clinical questions," *Journal of biomedical informatics*, vol. 44, no. 2, pp. 277–288, 2011.

[42] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, *et al.*, "An overview of the bioasq large-scale biomedical semantic indexing and question answering competition," *BMC bioinformatics*, vol. 16, no. 1, pp. 1–28, 2015.

[43] A. Nentidis, A. Krithara, K. Bougiatiotis, M. Krallinger, C. Rodriguez-Penagos, M. Villegas, and G. Paliouras, "Overview of bioasq 2020: The eighth bioasq challenge on large-scale biomedical semantic indexing and question answering," in *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings 11*, pp. 194–214, Springer, 2020.

[44] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[45] S. Sharma, H. Patanwala, M. Shah, and K. Deulkar, "A survey of medical question answering systems," *International Journal of Engineering and Technical Research (IJETR) ISSN*, vol. 3, pp. 131–133, 2015.

[46] M. Sarrouti and S. O. El Alaoui, "Sembionlqa: A semantic biomedical question answering system for retrieving exact and ideal answers to natural language questions," *Artificial intelligence in medicine*, vol. 102, p. 101767, 2020.

[47] S. J. Athenikos, H. Han, and A. D. Brooks, "A framework of a logic-based question-answering system for the medical domain (loqas-med)," in *Proceedings of the 2009 ACM symposium on Applied Computing*, pp. 847–851, 2009.

[48] X. Huang, J. Zhang, Z. Xu, L. Ou, and J. Tong, "A knowledge graph based question answering method for medical domain," *PeerJ Computer Science*, vol. 7, p. e667, 2021.

[49] T. Möller, A. Reina, R. Jayakumar, and M. Pietsch, "Covid-qa: A question answering dataset for covid-19," in *ACL 2020 Workshop on Natural Language Processing for COVID-19 (NLP-COVID)*, 2020.

[50] V. Liévin, C. E. Hother, and O. Winther, "Can large language models reason about medical questions?," *arXiv preprint arXiv:2207.08143*, 2022.

[51] G. Feng, Z. Du, and X. Wu, "A chinese question answering system in medical domain," *Journal of Shanghai Jiaotong University (Science)*, vol. 23, pp. 678–683, 2018.

[52] N. Jamali, Y. Yaghoobzadeh, and H. Faili, "Percqa: Persian community question answering dataset," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pp. 6083–6092, 2022.

[53] K. Darvishi, N. Shahbodaghkhan, Z. Abbasiantaeb, and S. Momtazi, "Pquad: A persian question answering dataset," *Computer Speech & Language*, vol. 80, p. 101486, 2023.

[54] R. AI, "Hazm: A python library for hazm: Persian text processing." `https://www.roshan-ai.ir/hazm/docs/`, 2021.

[55] N. Tohidi, C. Dadkhah, and R. B. Rustamov, "Optimizing persian multi-objective question answering system," *International Journal on Technical and Physical Problems of Engineering (IJTPE)*, vol. 13, no. 46, pp. 62–69, 2021.

[56] N. Taghizadeh, E. Doostmohammadi, E. Seifossadat, H. R. Rabiee, and M. S. Tahaei, "Sina-bert: a pre-trained language model for analysis of medical texts in persian," *arXiv preprint arXiv:2104.07613*, 2021.

[57] R. I. for Persian Studies, "Hazm: Persian nlp toolkit." `https://github.com/roshan-research/hazm`.

[58] I. Soboroff, "Overview of trec 2021," in *30th Text REtrieval Conference. Gaithersburg, Maryland*, 2021.

[59] C. Van Gysel and M. de Rijke, "Pytrec_eval: An extremely fast python interface to trec_eval," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 873–876, 2018.

[60] "Learn how to use trec_eval to evaluate your information retrieval system." `http://www.rafaelglater.com/en/post/learn-how-to-use-trec_eval-to-evaluate-your-information-retrieval-system`.

[61] Y. Zhang, Y. Wang, H. Zhang, B. Zhu, S. Chen, and D. Zhang, "Onelabeler: A flexible system for building data labeling tools," in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–22, 2022.

[62] H. Wu and N. Yuan, "An improved tf-idf algorithm based on word frequency distribution information and category distribution information," in *Proceedings of the 3rd International Conference on Intelligent Information Processing*, pp. 211–215, 2018.

[63] S. E. Robertson and S. Walker, "Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval," in *SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, organised by Dublin City University*, pp. 232–241, Springer, 1994.

[64] M. Ehrmann, A. Hamdi, E. L. Pontes, M. Romanello, and A. Doucet, "Named entity recognition and classification in historical documents: A survey," *ACM Computing Surveys*, vol. 56, no. 2, pp. 1–47, 2023.

[65] B. Mitra, N. Craswell, *et al.*, "An introduction to neural information retrieval," *Foundations and Trends® in Information Retrieval*, vol. 13, no. 1, pp. 1–126, 2018.

[66] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning–based text classification: a comprehensive review," *ACM computing surveys (CSUR)*, vol. 54, no. 3, pp. 1–40, 2021.

[67] A. Askari, S. Verberne, O. Alonso, S. Marchesin, M. Najork, and G. Silvello, "Combining lexical and neural retrieval with longformer-based summarization for effective case law retrieval.," in *DESIRES*, pp. 162–170, 2021.

[68] J. Lin, R. Nogueira, and A. Yates, *Pretrained transformers for text ranking: Bert and beyond*. Springer Nature, 2022.

Appendix A

width=!,height=!,pages=-,pagecommand=,width=