Title	Towards End-to-end Wikipedia-based Open-domain Question-Answering Systems for Single-hop and Multi-hop Questions in Low-resource Languages
Author(s)	Nguyen, Hien Dieu
Citation	
Issue Date	2023-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/18307
Rights	
Description	Supervisor: NGUYEN, Minh Le, 先端科学技術研究科, 修士(情報科学)

Master's Thesis

Towards End-to-end Wikipedia-based Open-domain Question-Answering Systems for Single-hop and Multi-hop Questions in Low-resource Languages

NGUYEN, Hien Dieu

Supervisor NGUYEN, Minh Le

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Master Degree)

March, 2023

Abstract

Open-domain Question-Answering (QA) task involves using a large knowledge base, such as Wikipedia, to answer a given question. This is often done using a two-stage framework that includes a Retriever and a Reader. The performance of the QA system is greatly influenced by the effectiveness of the Retriever stage. Despite being the first language of roughly a hundred million people worldwide, Vietnamese remains a low-resource language with a scarcity of research on QA systems. No efficient Vietnamese Open-domain QA system for single and multi-hop questions has been studied. Although resource-rich languages like English witnessed many advancements in Open-domain QA, these methods often suffer from low data situations. The objective of this study is to design an efficient Open-domain QA system utilizing the Wikipedia knowledge base, which can handle both single and multi-hop questions. The proposed system is robust when applied to low-resource languages. This research was initially conducted in the Vietnamese language, but the methodology can be generalized to other low-resource languages. This study proposes ViWiQA, an efficient Vietnamese Open-domain QA system over the Wikipedia knowledge base, with two novel retriever methods for single-hop and multi-hop questions. ViWiQA can be effectively trained with low data and significantly outperforms Lucene-BM25 and Dense Passage Retrieval when adapted to Vietnamese datasets. ViWiQA demonstrates a significant improvement of 20% in single-hop retrieval accuracy compared to Lucene-BM25 and sets a new standard in single-hop and multi-hop Vietnamese Open-domain QA benchmarks.

Acknowledgement

I am deeply grateful to my supervisor Professor NGUYEN Minh Le, my second supervisor Professor Satoshi Tojo, and my supervisor for minor research Professor Shinobu Hasegawa for their unwavering support, guidance, and mentorship throughout my academic journey. Their knowledge, experience, and insights have been instrumental in shaping this thesis and have been a constant source of inspiration. I am thankful for the endless hours of discussions, the encouragement during the tough times, and the constructive feedback that helped me improve my work.

I would like to extend my sincere appreciation to Japan Advanced Institute of Science and Technology (JAIST), for providing me with the resources, facilities, and opportunities that enabled me to pursue my research interests. I am grateful for the support of the faculty, staff, and administrators who have encouraged me throughout my studies.

I would also like to extend my sincerest appreciation to the Japanese Ministry of Education, Culture, Sports, Science, and Technology (MEXT) for providing me with the prestigious MEXT scholarship. This scholarship enabled me to pursue my academic interests and conduct my research in Japan. I am deeply grateful for the opportunity to experience the country's rich culture and education system and to learn from some of the world's finest minds. This scholarship will forever be a valuable experience and a treasured memory.

I would like to acknowledge the support of my family, friends, and colleagues, who have provided me with encouragement, motivation, and love during the long journey of writing this thesis. Their unwavering support and understanding have been a constant source of comfort and strength.

Finally, I would like to recognize the countless hours I spent reading, writing, and revising, and to acknowledge the many people who have helped shape this work. This thesis would not have been possible without their support and guidance, and I am grateful for the opportunity to acknowledge their contributions. Thank you for being a part of my academic journey.

Contents

1 Introduction 1

1.1 Background . . . 1

1.2 Objectives . . . 5

2 Related Works 8

2.1 Machine Reading Comprehension . . . 8

2.2 Information Retrieval . . . 9

2.3 Retriever-Reader Paradigm . . . 10

2.4 Question-Answering in Vietnamese . . . 10

3 Vietnamese Multi-hop QA dataset: VIMQA 12

3.1 Data Collection . . . 12

3.1.1 Wikipedia Graph . . . 13

3.1.2 Feasible Titles List . . . 13

3.1.3 Paragraph Pairs Selection . . . 14

3.1.4 Annotation by Crowd Workers . . . 14

3.1.5 Processing and Normalizing . . . 14

3.2 Data Analysis . . . 15

3.2.1 Question Analysis . . . 15

3.2.2 Answer Analysis . . . 15

3.2.3 Multi-hop Reasoning Type Analysis . . . 16

3.3 Benchmark Settings . . . 20

3.3.1 Data Splits . . . 20

3.3.2 Benchmark settings . . . 21

3.4 Experiments . . . 21

3.4.1 Experimental Settings . . . 21

3.4.2 Human Performance . . . 23

3.4.3 Results . . . 23

3.5 Summary . . . 25

4 Enhancing Reader performance by identifying relevant information 29
4.1 The Proposed System 29
4.1.1 Overall 29
4.1.2 Potential Sentence Classification Model 30
4.1.3 Answer Extraction 31
4.2 Experiments and Results 32
4.2.1 Dataset 32
4.2.2 Models 34
4.2.3 Experimental Results 35
4.3 Discussion 38
4.4 Summary 40

5 ViWiQA: Efficient End-to-end Vietnamese Wikipedia-based Open-domain Question-Answering Systems for Single-hop and Multi-hop Questions 42
5.1 Method 42
5.1.1 Problem formulation 42
5.1.2 Vietnamese Wikipedia Pre-processing 43
5.1.3 Retriever for Single-hop QA 43
5.1.4 Retriever for Multi-hop QA 44
5.1.5 Cross-Encoder model 46
5.1.6 Question-Answering Reader model 47
5.1.7 Training 48
5.1.8 End-to-end QA System 48
5.1.9 Wikipedia-Entity-Resolution and Model Ensemble . . . 49
5.2 Experimental Result 50
5.2.1 Experimental Setup 50
5.2.2 Results 52
5.3 Summary 62

6 Conclusions 66
6.1 Conclusions 66
6.2 Published Works 67
6.2.1 Related to Main research 67
6.2.2 Other publications 67

List of Figures

1.1 Typical processes of Question Answering systems 2
1.2 General pipeline of Open-domain single-hop and multi-hop QA systems 7

3.1 Overall data collecting pipeline of VIMQA 26
3.2 User interface for annotators to input a sample 27
3.3 Percentage of question types in VIMQA 27
3.4 Distribution of question lengths in VIMQA 28

4.1 An overview of proposed system 30
4.2 Comparing the number of tokens in the original and reduced contexts 36

5.1 Overall of the proposed single-hop retriever component of ViWiQA-Single 44
5.2 Overall of the proposed multi-hop retriever component of ViWiQA-Multi 45
5.3 Detailed architecture of Cross-Encoder model for predicting paragraph relevance scores 46
5.4 Detailed architecture of Reader model for predicting answer spans 63
5.5 Overall end-to-end QA system with Wikipedia entity resolution. 64
5.6 Ensemble approach for multiple retrievers and readers. 64
5.7 Retrieval accuracy of DPR on UIT-ViQuAD test set at different training batch sizes when adapted to Vietnamese Wikipedia, measured by the proportion of top k retrieval results containing the answer. 65

List of Tables

3.1 Vietnamese Central Question Words Collection 16

3.2 Types of answers in VIMQA 17

3.3 Classification of the reasoning necessary to answer questions in VIMQA, including English interpretations in *italics*. The linking entity is displayed in **bold orange**. Supporting details for answers are shown in blue. The answers themselves are emphasized in **bold green**. Words representing the reasoning type are marked in **purple**. 20

3.4 Result of 5-fold cross-validation on VIMQA 20

3.5 VIMQA's data division 21

3.6 Performance assessment of evaluated method on dev and test sets of VIMQA under two benchmark configurations 22

3.7 Evaluation of models' capabilities on VIMQA under the Gold Only setting and comparison with UIT-ViQUAD 24

3.8 Evaluating existing methods using three metrics on the Distractor test set of VIMQA 24

4.1 The detailed analysis of the datasets in the experiments. . . . 33

4.2 Overview of the generated datasets for PSCM. 33

4.3 Result of the PSCM module evaluation on the generated dataset 35

4.4 Result on Qasper dataset of single model and our method . . . 36

4.5 Result on UIT-ViQuAD dataset of single model and our method 37

4.6 Compares our method(applying on XLM-RoBERTa) and the ViReader. The ViReader depends on the numbers of sentences (K) in the retrieval step 37

4.7 Result on SQuAD dataset of single model and our method . . 38

4.9 Compares the context reduced using our pipeline and using ViReader retrieval module. The correct answer is highlight in red 39

4.8 Example in SQuAD 2.0 where distracting information affects the model decision. The correct answer is highlight in red . . . 41

5.1 Overall of Vietnamese datasets for single-hop QA (UIT-ViQuAD) and multi-hop QA (VIMQA) 50
5.2 Comparison of Vietnamese Wikipedia and UIT-ViQuAD knowledge base 51
5.3 Top-k retrieval accuracy on UIT-ViQuAD development and test sets using Vietnamese Wikipedia knowledge base, measured by the proportion of top k passages retrieved containing the answer. 51
5.4 End-to-end single-hop QA performance on UIT-ViQuAD using the Wikipedia knowledge base. Lucene-BM25 and DPR are adapted to Vietnamese Wikipedia from the cited papers. The Reader model of these systems is XLM-RoBERTa$_{\text{Large}}$. . . 53
5.5 End-to-end QA performance on UIT-ViQuAD test set using UIT-ViQuAD knowledge base. 53
5.6 Multi-hop retrieval accuracy over the Wikipedia knowledge base, evaluated on VIMQA dataset, measured in the percentage of retrieved passage pairs that have at least one correct passage (1C), has two correct passages (2C) or contains the answer (CA). Yes/No questions are excluded when measuring CA accuracy. 54
5.7 End-to-end evaluation of multi-hop QA systems on VIMQA measured in Exact Match (EM) and F1 Score 55
5.8 Evaluation of ViWiQA-ER on Zalo AI Challenge 2022 (End-to-end QA task) public test set. 56
5.9 Comparing different Vietnamese and multilingual pre-trained Transformer models on UIT-ViQuAD test set (QA with given contexts task). 56
5.10 Top-k retrieval accuracy on UIT-ViQuAD development and test sets using Vietnamese Wikipedia knowledge base, measured by the proportion of top k retrieval results containing the answer. ViWiQA-Single is evaluated in three settings: Only rerank using the Cross-Encoder (CE), only filter by the predictions of Reader (RE), and full method using both the CE scores and Reader predictions (Full). See text for more details. 57
5.11 End-to-end single-hop QA performance on UIT-ViQuAD using the Wikipedia knowledge base. ViWiQA is evaluated using only the Cross-Encoder (CE), only Reader predictions (RE), and full method (Full). 57
5.12 Example 1: Passage-pair retrieval using Cross-Encoder and ViWiQA-Multi. English translation is provided in *italic.* . . . 60

5.13 Example 2: Passage-pair retrieval using Cross-Encoder and ViWiQA-Multi. English translation is provided in *italic*. . . . 62

Chapter 1

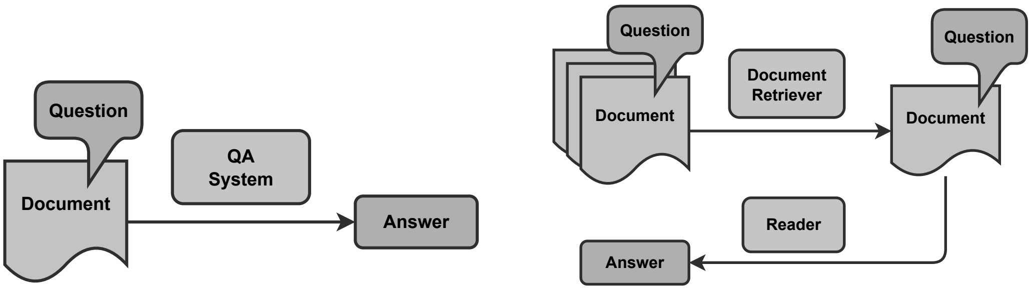
Introduction

1.1 Background

Question Answering (QA) is one of the core disciplines within information retrieval in general and natural language processing in specific. It has lately gained more attention in the research community as well as the enterprise. The goal of QA systems is to automatically answer human questions in a natural language from the given context. In particular, a sample of a QA model often is a pair of a given sequence and a question. Therefore, QA systems require the text understanding of natural language to find the relationship between contexts and questions. However, inputs often contain a lot of redundant information, which is useless for answering. The key research question in most QA systems is how to determine critical sentences and eliminate redundancy.

Based on the complexity of the input, QA systems can be divided into traditional QA systems and modern QA systems. In traditional QA systems [22, 55], the input is a single document or passage and a question. The system aims to extract the answer to the question from the document. Figure 1.1a illustrates the simple process of a traditional QA system. In modern QA systems [5, 45], the input contains a collection of documents and a question. Therefore, a typical modern QA system is usually a 2-step process. The first step is the retrieval phase aiming to find the relevant documents. The second step is text understanding, where the reader's goal is to extract answers from the relevant documents. Figure 1.1b shows the 2-step process of a typical QA system.

Open-domain QA [58] is an essential QA task requiring the system to input a question and seek the answer using a knowledge base. Early QA approaches are often sophisticated due to many components constituting the



(a) Traditional Question Answering system process. The input is a document containing the answer, and a question. The output is the answer to the question

(b) Modern Question Answering system process. The input is a collection of documents containing the answer, and a question. The output is the answer to the question

Figure 1.1: Typical processes of Question Answering systems

system ([18, 36]). As deep learning progress, more recent methods take advantage of Machine Reading Comprehension (MRC) approaches and simplify the system into a framework consisting of two components: Retriever and Reader. For a query, the Retriever aims to retrieve relevant documents from the knowledge base, and the Reader aims to find the answer using those documents. Advancements in Open-domain QA are witnessed in resource-rich languages like English, with many retriever methods proposed. While sparse retrievers like TF-IDF or BM25 were used in early QA systems [6], retriever approaches using dense representations [25, 31, 21, 53, 19] produced competitive results and became a new paradigm for passage retrieval in Open-domain QA. Using a reranker to rerank the retrieval result was also a popular technique in Information Retrieval (IR) ([44, 46]) and Open-domain QA ([60, 30, 34]). Many datasets for QA and Open-domain QA were proposed in English ([49, 24, 27, 4, 2]). Figure 1.2a illustrates a system of Open-domain single-hop QA with a general pipeline. The task of multi-hop QA, which needs the system to relate pieces of information from multiple documents, was also proposed. Yang et al. [69] proposed HotpotQA as a large-scale multi-hop dataset and provided benchmarks for QA with given contexts and Open-domain QA. Multi-hop retrievers that aim to retrieve the passage pairs to perform multi-hop reasoning were proposed ([1, 65]). Figure 1.2b shows a general pipeline of Open-domain multi-hop QA systems.

One of the core components in QA systems is Machine Reading Comprehension (MRC) as the reader. Machine Reading Comprehension refers to the machine's ability to read, comprehend a given text passage, and answer questions based on it. MRC has increasingly attracted interest in the research community on natural language understanding. The MRC task is proposed as a QA problem where the system automatically extracts answers to questions from a given document. Another essential component that decides a QA system's performance is the Information Retrieval (IR) module. IR refers to the process of retrieving information resources that are relevant to a query from a collection of passages. In a modern QA system, the input is a list of documents and a question. The length of the input documents is remarkably challenging in modern systems. Therefore, a modern QA system usually has an IR component to extract the relevant documents before extracting the answer via MRC component. In addition, previous works [17, 26] have shown that the performance of the machine reading comprehension component can be improved using summarization. It reveals the potential and necessity of IR in modern QA systems where the input information is more massive and diverse.

Distracting information in the context can be a significant factor that reduces the QA model's performance. However, it is still a challenging and ambitious goal in many existing QA approaches. Nguyen et al.[38] proposed ViReader, which employs a phase to select *top-k* sentences in the context that are similar to the question and achieves state-of-the-art performance on Vietnamese QA datasets UIT-ViQuAD[40]. However, this method is constrained by a fixed number k for every context. As a result, it is limited to improve the QA model because different contexts have distracting information with different sizes.

Although many state-of-the-art methods in Open-domain QA achieved outstanding performances in English QA benchmarks, their capabilities are not guaranteed when adapted to a low-resource language like Vietnamese. Retriever approaches using dense representation like Dense Passage Retrieval (DPR) ([25, 65]) were shown to require much data and complex techniques for efficient training and perform poorly in low data circumstances ([19, 53]). This behavior can be observed in our experiments when adapting DPR to Vietnamese datasets. While Cross-Encoder rerankers ([14, 68]) are more effective than dense retrievers in low data situations, they are impractical when used with a large number of documents in a corpus regarding one question. Moreover, it is not trivial to derive the training data for Cross-Encoder from a QA dataset. Based on related works, we believe that no effective retriever method has been suggested for single or multi-hop Vietnamese Open-domain QA. The reader component also plays an essential role in Vietnamese QA systems. Common Vietnamese reader approaches ([39, 43]) used the pre-trained multilingual model XLM-RoBERTa [10] and showed potential results. No study has been conducted to compare reader approaches using different pre-trained models on Vietnamese corpus like PhoBERT [37], and multilingual corpus such as Multilingual-BERT, XLM-RoBERTa. Besides, only a handful of datasets and studies were proposed for Vietnamese QA. UIT-ViQuAD [41] is only large-scale Vietnamese dataset for single-hop QA. Nguyen et al. [43] proposed a Vietnamese Open-domain single-hop QA system XLMRQA. However, XLMRQA only uses a simple sparse retriever, and its knowledge base is about 800 times smaller than the Vietnamese Wikipedia in the number of passages. To our knowledge, no efficient Vietnamese Open-domain QA system for single and multi-hop questions has been proposed. Also, no QA system that uses the entire Vietnamese Wikipedia as the knowledge base has been proposed.

Entity resolution (ER) is the task of identifying entities that refer to the same real-world entity across different data sources. Regarding End-to-end Open-domain QA systems, the entity that can answer the question might have different aliases across the knowledge base. For example, on Wikipedia, "Edson Arantes do Nascimento" and "Pelé" can refer to the same professional footballer. This creates inconsistency in the responses of QA systems, where the answer of the systems vary depending on the retrieved contexts and knowledge bases. Moreover, answers from the QA system can be ambiguous and refer to multiple entities. For example, in Wikipedia 2022, the entity "James Abbott" referred to 8 different persons, and the entity "La Villa" referred to 8 different places. It is essential that the QA systems can provide the user with the correct entity. However, the problem of ER in Open-domain QA is overlooked, and no research has been conducted on the Vietnamese knowledge bases. The contest Zalo AI Challenge 2022 [1] introduced the End-to-end QA task over Wikipedia, where the input is a plain text question, and the output answer can be a Wikipedia entity, a specific date, or a number. This task posed many challenges to existing QA approaches.

The task of Open-domain Question-Answering (QA) [58] involves using a large knowledge base, such as Wikipedia, to answer a given question. Contemporary QA systems often employ a two-stage framework called Retriever-Reader [7, 31, 25, 21, 53], where the performance of the system is heavily influenced by the efficiency of the Retriever stage. Vietnamese, despite being the native language of over 98 million people worldwide, is considered a low-resource language with limited research on QA systems [42, 39]. There is currently no efficient system for answering both single and multi-hop questions in Vietnamese. While languages like English have seen notable advancements in Open-domain QA, these methods often struggle with low data situations. Furthermore, existing Vietnamese QA datasets do not assess the model's ability to perform advanced reasoning and provide explanations for the an-

[1] An annual AI competition in Vietnam (challenge.zalo.ai)

swer.

1.2 Objectives

The objective of this research is to create an effective Open-domain QA system that utilizes the Wikipedia knowledge base for answering both single and multi-hop questions. The proposed system is robust when applied to low-resource languages. This research was initially conducted in the Vietnamese language, but the methodology can be generalized to other low-resource languages. To this end, the research has the following contributions:

1. This research unveils VIMQA, a novel Vietnamese dataset comprising more than 10,000 Wikipedia-based multi-hop question-answer pairs.

2. The research proposes a novel pipeline to enhance the performance of existing Reader models by identifying relevant information from the context.

3. The research proposes ViWiQA, an efficient Vietnamese Open-domain QA system over the Wikipedia knowledge base, with two novel retriever methods for single-hop and multi-hop questions.

This research specifically presents the following contributions through the VIMQA dataset:

1. The introduction of VIMQA, a new Vietnamese dataset that focuses on advanced reasoning and providing explainable answers to multi-hop questions.

2. The development of a framework for collecting multilingual multi-hop question-answer datasets, originally tailored for the Vietnamese language.

3. An in-depth analysis of different linguistic aspects of the dataset.

4. The evaluation of the dataset through current baselines and state-of-the-art methods in question-answering to showcase its quality and robustness.

Additionally, the ViWiQA system makes the following contributions:

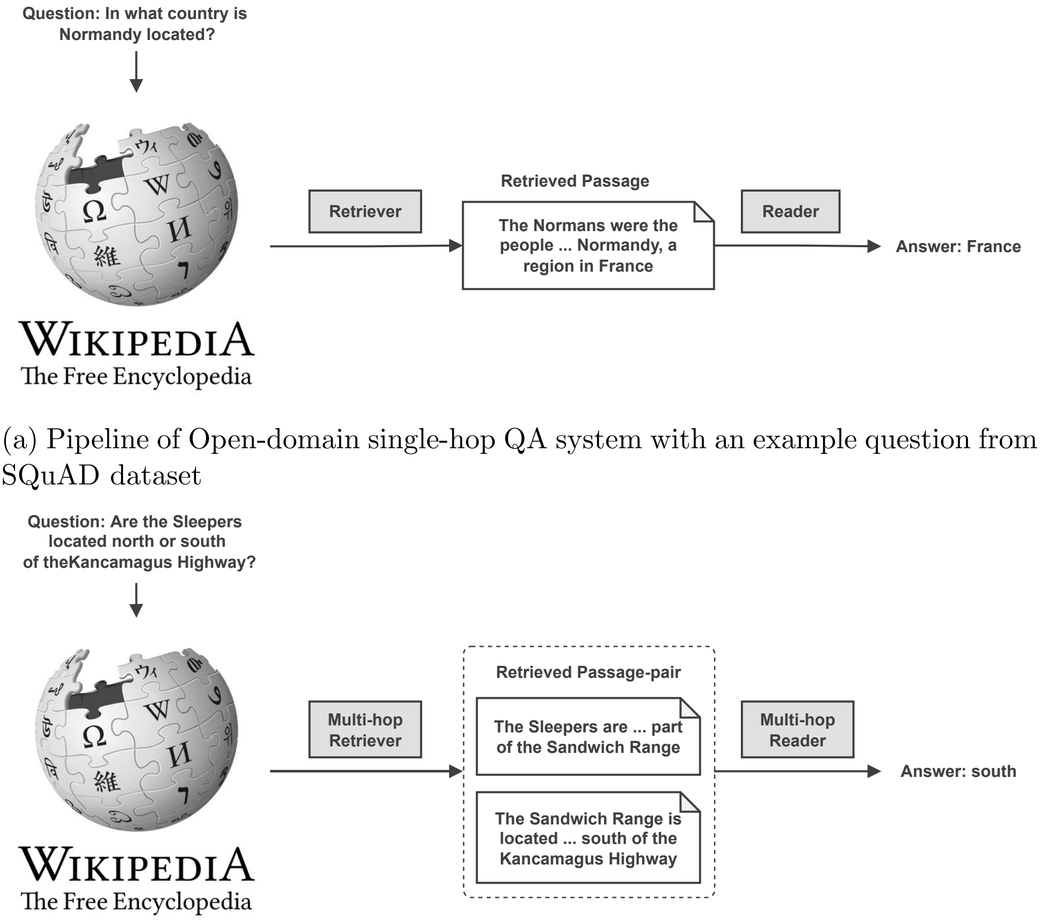
1. The development of a retriever method for single-hop Vietnamese Open-domain QA that can be efficiently trained with low resources and establishes state-of-the-art retrieval accuracy.

2. The proposal of a technique for retrieving passages for multi-hop Vietnamese Open-domain QA that utilizes a graph constructed from Wikipedia hyperlinks, resulting in state-of-the-art performance.

3. The introduction of the ViWiQA system, which incorporates the proposed retriever methods and an effective reader model, and achieves the highest performance on standard benchmarks for single and multi-hop Vietnamese QA.

4. The presentation of a straightforward method for addressing entity resolution in Wikipedia knowledge bases. Also, the development of an end-to-end QA system for the Vietnamese Wikipedia, which generates specific Wikipedia entities as answers instead of plain text. The approach was successful in achieving the **2$^{\text{nd}}$ Place** in the Zalo AI Challenge 2022 Contest.

5. The ablation study that analyzes essential aspects when adapting state-of-the-art English methods to low-resource situations like Vietnamese. The adaptation of these methods and the proposed ViWiQA make a solid foundation for future research in Vietnamese and multilingual Open-domain QA.


Question: In what country is Normandy located?
WIKIPEDIA
The Free Encyclopedia
Retriever
Retrieved Passage
The Normans were the people ... Normandy, a region in France
Reader
Answer: France
Question: Are the Sleepers located north or south of theKancamagus Highway?
WIKIPEDIA
The Free Encyclopedia
Multi-hop Retriever
Retrieved Passage-pair
The Sleepers are ... part of the Sandwich Range
The Sandwich Range is located ... south of the Kancamagus Highway
Multi-hop Reader
Answer: south

(a) Pipeline of Open-domain single-hop QA system with an example question from SQuAD dataset

(b) Pipeline of Open-domain multi-hop QA system with an example question from HotpotQA dataset

Figure 1.2: General pipeline of Open-domain single-hop and multi-hop QA systems

Chapter 2

Related Works

2.1 Machine Reading Comprehension

Machine Reading Comprehension (MRC) is a subfield of Artificial Intelligence (AI) that focuses on developing systems that can understand natural language text. The goal of MRC is to enable computers to answer questions that are based on the contents of a given text. MRC systems typically consist of a text encoder that converts the input text into a machine-readable format, and a question encoder that converts the question into a machine-readable format. The encoded text and question are then passed through a matching module, which extracts the relevant information from the text to answer the question.

In previous studies, MRC methods can be broadly classified into two categories: Traditional Neural Network and Transformer. Initially, with the advent of high-quality datasets, a number of MRC models were developed using neural networks. These models demonstrated outstanding performance on common MRC datasets and were found to be more durable than traditional machine-learning methods that use handcrafted features. Notable examples of this category include Match-Long Short Term Memory [59], R-Net [20], DrQA Reader [5], FastQA [62], Bi-directional Attention Flow [56], QANet [71], and FusionNet [47].

On the other hand, the achievement of the Transformer model in NLP has had a significant impact on various areas, including MRC. Indeed, many Transformer-based models have demonstrated their effectiveness in a wide range of NLP tasks and applications. Recently, models such as BERT [15], XLM-R [11], and ALBERT [28], which are variations of the Transformer model, have set new benchmarks on MRC datasets. The power of these approaches comes from pre-trained parameters in large datasets. Therefore, to take advantage of these portable language models, we also incorporate them in our MRC phase.

2.2 Information Retrieval

Information Retrieval (IR) is the process of obtaining information that is relevant to a user's needs from a collection of data sources. It is a subfield of computer science and information science that deals with the process of retrieving information from a collection of documents or databases. The goal of IR is to provide a set of relevant documents or information in response to a user's query. IR systems are used in a wide range of applications, including web search engines, digital libraries, and enterprise search.

Based on the type of learning, IR systems can be broadly categorized into two types based on the type of learning: supervised and unsupervised. Supervised information retrieval (SIR) is based on labeled data, where the relevant documents are already known and labeled. The goal of SIR is to learn a model that can predict the relevance of new documents based on the labeled data. Examples of supervised learning techniques used in information retrieval include Support Vector Machines (SVMs) [23], Logistic Regression, and Neural Networks. On the other hand, unsupervised information retrieval (UIR) is based on unlabeled data, where the relevant documents are not known. The goal of UIR is to discover patterns in the data that can be used to identify relevant documents. Initially, unsupervised methods commonly use frequency and probability-based features like TF-IDF, BM25 [54], and TextRank [35].

Besides, IR using transformer models is a recent trend in the field of natural language processing. These models are based on the transformer architecture, which was first introduced in the paper "Attention Is All You Need" by Vaswani et al. [57]. Transformer models have been shown to be highly effective in a wide range of NLP tasks. In IR, transformer models are used to encode the query and document text, and the attention mechanism is used to align the query with the relevant parts of the document, making it possible to retrieve relevant documents from a large corpus. An example of transformer-based IR models is Dense Passage Retrieval (DPR) [25]. These models have achieved state-of-the-art results on various IR benchmarks and have shown to be highly effective in improving the effectiveness of retrieval.

2.3 Retriever-Reader Paradigm

The Retriever-Reader pipeline, which is widely used for many QA tasks, is one of the commonly employed techniques for Open-domain Question Answering [72]. Sparse retrievers, which utilize sparse vector space models such as TF-IDF or BM25, are commonly used for QA tasks [6, 67]. In contrast, dense retrievers, which represent questions and documents as dense vectors through dual-encoding, are also commonly used [25, 21, 19, 53]. Another line of work focuses on reranking the passages retrieved in the first-stage retriever. Cross-Encoder[14] used for Rerankers in Open-domain QA has shown substantial enhancement ([44, 60, 66]). Reranking passages using predictions from the reader model also shows potential improvement [34].

The reader component can be classified into two types: extractive and generative. Extractive readers aim to identify the most relevant segments from the provided documents as answers, while generative readers aim to generate answers through sequence-to-sequence techniques. Earlier QA systems often employ Extractive Reader [6, 60, 25, 21], and some recent systems use Generative Reader [32, 65].

The Retriever-Reader pipeline is also widely used in multi-hop Open-domain QA. For multi-hop questions, the model needs to perform reasoning over multiple documents. As a result, the retriever component typically retrieves multiple passages instead of one. In the HotpotQA dataset [69], the questions require reasoning over two passages. In many multi-hop QA systems ([69, 1, 65]), the number of passages retrieved by the retriever is also two.

2.4 Question-Answering in Vietnamese

The field of QA and Open-domain QA in Vietnamese has not seen much research due to its low-resource nature. As far as we know, the only large-scale dataset for single-hop QA in Vietnamese is UIT-ViQuAD [42]. A few Vietnamese QA systems have been developed, such as ViReader [39], which is an MRC system that uses given contexts, and XLMRQA [43], which is a system for Open-domain QA that utilizes all passages in ViQuAD as the knowledge base.

However, no effective retriever method for single and multi-hop Vietnamese Open-domain QA has been researched. Additionally, no Vietnamese QA systems that use large-scale knowledge bases like the entire Wikipedia have been studied. This lack of research in Vietnamese QA highlights the need for further investigations in this field, especially in the areas of effective retriever methods for Open-domain QA and the use of large-scale knowledge bases. Such research would be beneficial in developing more accurate and practical QA systems for Vietnamese.

Chapter 3

Vietnamese Multi-hop QA dataset: VIMQA

This chapter presents the Vietnamese Multi-hop Question Answering Dataset (VIMQA), which is designed to test the ability of QA systems to perform multi-hop reasoning and provide supporting facts to guide the inference process. The chapter also proposes an effective method and framework for collecting VIMQA through crowdsourcing using Wikipedia articles. To ensure that the questions in VIMQA are natural and not constrained to any pre-existing knowledge base, crowd workers were shown multiple supporting paragraphs and asked to generate questions that required reasoning across all of the paragraphs. They were also asked to provide the answers and evidence in the paragraphs that support the answers. VIMQA dataset is publicly available on the website `https://github.com/vimqa/vimqa`.

3.1 Data Collection

In this section, we outline our data collection pipeline. Based on Yang et al.'s work [70], our aim is to create a framework for collecting multilingual, explainable QA datasets that require multi-hop reasoning. Our framework is primarily used for Vietnamese but can be adapted to other languages. Despite the existence of some multi-hop QA datasets, our framework offers convenience and simplicity in multi-hop QA development.

Traditionally, multi-hop datasets are collected through reasoning chains using a knowledge base, but this approach may result in limited diversity [70]. Inspired by Rajpurkar et al. [50] and Yang et al. [70]'s text-based QA dataset collection, we design a similar framework with minimal modifications. A typical QA sample includes context and a question, where the answer must be extracted from the context and the question requires multi-hop reasoning across multiple contexts.

Our target dataset requires advanced reasoning over multiple paragraphs and the ability to provide supporting facts for explainable predictions. The data collection pipeline should also be flexible and easily adaptable to any language.

We present a data collection pipeline in Figure 3.1. The process begins by selecting a title randomly from a list of suitable options. From there, a paragraph pair is randomly selected from Wikipedia graph using the chosen title. Crowd workers then create questions, answers, and supporting information based on the pair. The resulting annotated sample is then cleaned and standardized through our configuration process. Further details of each component and step are covered in subsequent sections.

3.1.1 Wikipedia Graph

Our proposed framework, VIMQA, is applied on the Vietnamese Wikipedia. It shares similarities with the English Wikipedia, as noted by Yang et al. [70]: the hyperlinks in Wikipedia articles are useful for multi-hop reasoning, and the summaries of articles contain the most important information. Thus, we treat the Vietnamese Wikipedia as a directed graph where each vertex is a unique article, represented by its title, and each edge between two vertices represents a hyperlink. The summary of each article is the only part we focus on.

3.1.2 Feasible Titles List

The Vietnamese Wikipedia has around 1.2 million articles, smaller than the English Wikipedia by about four times. However, not all articles are suitable for creating multi-hop questions. Some general concepts, like "football," "city," and "music," are hard to create multi-hop questions from. In contrast, articles about specific people, events, or places are easier to create questions from. Technical articles, such as "Binary search tree" and "TCP/IP," can also pose difficulties in creating meaningful questions. To tackle this, we manually selected a list of suitable article titles that are straightforward to create multi-hop questions from. Although a tool is provided to collect all titles from the Wikipedia, users should narrow down the list to their specific needs.

3.1.3 Paragraph Pairs Selection

To generate questions that require multi-hop reasoning, crowd workers are provided a pair of paragraphs. Our paragraph selection process mirrors that of HotpotQA [70]. For example, to answer the question, “Where is the club John O’Shea joined when he was 17 based?”, multi-hop reasoning is necessary to determine that “Manchester United” was the club O’Shea joined and then locate where it is based. The “Manchester United” in this example can be viewed as the bridge entity connecting the two paragraphs. To obtain paragraph pairs, we first randomly select a title A from the list of feasible titles and then choose an edge (A, B) in the Wikipedia graph where B is also a feasible title. The paragraphs from A and B are then given to the crowd workers to create QA data.

To generate questions for comparisons between entities of the same category, we create lists of similar entities, such as “Footballers”, “Musicians”, “Scientists”, “Organizations”, and “Countries”. To sample a pair of paragraphs for comparison questions, two paragraphs from the same list are randomly selected and given to the crowd worker for QA data creation. This type of question creation, as observed in HotpotQA [70], results in interesting questions, such as “Does Cristiano Ronaldo have more titles than Ryan Giggs?”

3.1.4 Annotation by Crowd Workers

To create a QA sample, the crowd worker must supply a multi-hop question, answer, and relevant facts, using a pair of paragraphs as a reference. We have created a user-friendly interface for the crowd workers to perform this task, as shown in Figure 3.2. The interface provides clear instructions and only accepts submissions that meet all requirements to minimize human error. The crowd workers are also reminded that multi-hop questions can be created by inquiring about the bridge entity.

Three researchers fluent in Vietnamese were tasked with annotating the VIMQA dataset. At the end of each day, they reviewed each other’s examples. Only examples verified by multiple workers were included in the dataset.

3.1.5 Processing and Normalizing

The Vietnamese language has unique characteristics that require different processing and normalization. One issue with normalizing Vietnamese is the Unicode encoding of accents, where accented letters like “á” can be encoded using either one Unicode point (U+00E1) or two Unicode points (combining acute accent - U+0301 and lower case letter A - U+0061). This is due to the complex tonal symbols used in Vietnamese. Since the data was collected from crowd workers, the encoding depends on their software, which can lead to different interpretations by computer models. Our dataset solves this issue by normalizing all accented Vietnamese letters to a single Unicode point.

Normalizing accent position in Vietnamese words is important due to the impact it can have on computer interpretation. For instance, "hoà" and "hòa" may look similar to humans but mean different things. To address this, we normalize words based on official dictionaries.

Post-processing and normalization require consideration of the specific traits of a language. While we have equipped tools for Vietnamese, it's up to users to modify and tailor them to their language. Our framework is highly adaptable and can be adjusted with minimal modifications for any language.

3.2 Data Analysis

3.2.1 Question Analysis

In our analysis, we examine the typical length and types of questions in the VIMQA dataset. We identify the different question types and create a list of central question words in Vietnamese to categorize the questions, as shown in Table 3.1. Questions not found in the CQW list are manually classified into eight broader categories.

We analyzed the distribution of question length and types in our VIMQA dataset. First, we identified various question types in the dataset and defined a list of central question words (CQW) in Vietnamese to categorize them, as shown in Table 3.1. Questions that don't belong to the CQW list are manually classified into eight main categories. The distribution of question types is shown in Figure 3.3, where Yes/No questions make up around a third of all questions. Additionally, "What", "Which", and "Who" questions are the most prevalent, similar to what was observed in HotpotQA [70].

We also examine the distribution of question length in the VIMQA dataset. Figure 3.4 displays the distribution of question lengths and it is clear that questions vary significantly in size.

3.2.2 Answer Analysis

Our analysis also looks at the distribution of answer types in the VIMQA dataset by sampling 100 examples, similar to the configuration of HotpotQA

Group	English CQW	Vietnamese CQW
Yes/No	Copulas (is, are) Aux (does, did)	Phải không, Đúng không
Which	Which	Nào
What	What	Là gì
	What ordinal number	Thứ mấy, Thứ bao nhiêu
Who	Who	Ai
	By whom	Bởi ai
How	How many	Bao nhiêu
	How often	Bao lâu một lần
	How long	Bao lâu
	How far	Bao xa
When	When	Khi nào
Where	Where	Ở đâu, Tại đâu
Why	Why	Vì sao, Tại sao

Table 3.1: Vietnamese Central Question Words Collection

in English. Table 3.2 displays the answer types. The answer type distribution shows that the VIMQA dataset includes a wide range of answers, which supports the findings from the analysis of question types. The largest categories of answers are Yes/No (28%), location (15%), date/time (12%), and person (11%). This demonstrates that the VIMQA dataset is of high quality and offers challenging multi-hop QA for the Vietnamese language.

3.2.3 Multi-hop Reasoning Type Analysis

To gain a deeper understanding of the various forms of multi-hop reasoning in VIMQA, we hand-classified 100 randomly selected examples from the dev and test sets. Table 3.3 presents the categories of reasoning along with examples.

Type I reasoning, also known as chain reasoning, requires identifying a bridge entity in the question and its location in the context, followed by second-hop reasoning to answer the question. This type is most prevalent in the dataset.

Type II involves determining the correct entity from a list based on checking multiple properties of the entity.

Type III calls for more complex inference using more than two supporting facts, whereas Type IV requires comprehension of the properties of two entities in the question.

Answer Type	%	Example(s)
Yes/No	28	Đúng, Không *(Yes, No)*
Location	15	Nhật Bản, Anh *(Japan, England)*
Date and time	12	1908, thời kỳ trị vì của Trần Nhân Tông *(1908, the reign of King Tran Nhan Tong)*
Person	11	Benjamin Franklin, Nguyễn Phú Trọng
Group / Org	6	The Beatles, Republic Records
Title / Nick name	5	Ông hoàng nhạc pop, Quỷ Đỏ *(King of Pop, Red Devils)*
Ordinal Number	4	hạng nhất, hạng tư *(first prize, fourth prize)*
Number	8	130 triệu; 45,5 tỷ bảng Anh *(130 million, 45.5 billion pounds)*
Proper noun	6	I'm Too Sexy, dân tộc Nùng *(I'm Too Sexy, Nung ethnic group)*
Common noun	3	hoá học, rắn hổ mang chúa *(chemistry, King cobra)*
Other	2	bằng thiết bị kết nối Internet *(with an Internet-connected device)*

Table 3.2: Types of answers in VIMQA

Also, we created a new type of question (Type V) that tests the ability to recognize negation and false entities in the context. This type is a Yes/No question that requires identifying negation/entity swap to answer yes or no, and thus necessitates multi-hop reasoning.

Reasoning Type	**%**	**Example(s)**

I. Inferring the **bridge entity** to complete the 2nd-hop question	54	**Question:** Đạo diễn phim It Happened One Night sinh ra ở đâu? *(Where was the director of It Happened One Night born?)* **Paragraph 1:** It Happened One Night là một bộ phim hài Mỹ ..., đạo diễn **Frank Capra**. *(It Happened One Night is a comedy film ..., directed by **Frank Capra**)* **Paragraph 2:** **Frank Capra** ... Sinh ra ở **Ý** và lớn lên ở Los Angeles ... *(**Frank Capra** ... Born in **Italy** and raised in Los Angeles ...)*
II. Locating the **answer entity** by checking multiple properties	28	**Question:** David Crosby từng là thành viên sáng lập của ban nhạc nào tan rã vào năm 1973? *(Which band did David Crosby founded broke up in 1973?)* **Paragraph 1:** David Van Cortlandt Crosby ... còn là thành viên sáng lập của các ban nhạc **The Byrds**, Crosby, Stills & Nash ... *(David Van Cortlandt Crosby ... was also a founding member of **The Byrds**, Crosby, Stills & Nash ...)* **Paragraph 2:** **The Byrds** là ban nhạc rock ... cho tới khi tuyên bố tan rã vào năm 1973. *(**The Byrds** were a rock band ... until their disbandment in 1973.)*
III. Other types of reasoning that require more than two supporting facts	4	**Question:** Giải đấu nào Fabien Barthez từng có một số danh hiệu được điều hành bởi Ligue de Football Professionnel? *(Which league did Fabien Barthez have several titles is run by the Ligue de Football Professionnel?)* **Paragraph 1:** Fabien Alain Barthez ... đã từng chiến thắng tại giải Cúp các đội vô địch bóng đá quốc gia châu Âu, một số danh hiệu tại **Giải vô địch bóng đá Pháp** và Giải bóng đá Ngoại hạng Anh. *(Fabien Alain Barthez ... has won the UEFA Champions League, several titles at **The French national football championship** and The English Premier League.)* **Paragraph 2:** **Giải bóng đá vô địch quốc gia Pháp** (tiếng Pháp: **Ligue 1**), ... Được điều hành bởi Ligue de Football Professionnel, Ligue 1 bao gồm ... *(**The French national football championship** (French: **Ligue 1**), ... Administrated by the Ligue de Football Professionnel, Ligue 1 consists of ...)*

IV. Comparing **two entities**	7	**Question:** **Daniel Sturridge** và **Frank Lampard** đều có chơi cho câu lạc bộ Chelsea phải không? *(Do **Daniel Sturridge** and **Frank Lampard** both play for Chelsea Football Club?)* **Answer:** **đúng** ***(yes)*** **Paragraph 1:** **Daniel Andre Sturridge** ... Anh rời Manchester City ... và gia nhập Chelsea theo dạng chuyển nhượng tự do. *(**Daniel Andre Sturridge** ... He left Manchester City ... and joined Chelsea as a free agent.)* **Paragraph 2:** **Frank James Lampard OBE** ... Anh được xem là một trong những cầu thủ xuất sắc nhất lịch sử của Chelsea và ... *(**Frank James Lampard OBE** ... He is considered to be one of Chelsea's greatest ever players and ...)*
V. Identifying the **Negation** factor to answer Yes/No questions	4	**Question:** Francesco Totti **chưa từng** thi đấu cho đội bóng nào ở Ý phải không? *(Have Francesco Totti **never** played for any Italian football club?)* **Answer:** **không** ***(no)*** **Paragraph 1:** Totti giải nghệ ngày 28 tháng 5 năm 2017 sau khi cùng **Roma** giành chiến thắng 3-2 trước Genoa ... *(Totti retired on May 28th 2017 after playing for **Roma** in a 3-2 win over Genoa ...)* **Paragraph 2:** **A.S. Roma** ... là một đội bóng thủ đô của Ý, ... *(**A. S. Roma** ... is an Italian capital professional football club, ...)*
VI. Identifying the **Entity** Swap to answer Yes/No questions	3	**Question:** Đội bóng của Nathan Dyer thành lập năm **1812** phải không? *(Was Nathan Dyer's football team founded in **1812**?)* **Answer:** **không** ***(no)*** **Paragraph 1:** Nathan Antone Jonah Dyer ... hiện đang chơi cho đội **Swansea City** ở vị trí tiền vệ cánh. *(Nathan Antone Jonah Dyer ... currently plays for **Swansea City** as a midfielder.)* **Paragraph 2:** **Swansea City** Association Football Club (thành lập năm **1912**) là một câu lạc bộ bóng đá chuyên nghiệp có trụ sở tại ... *(Swansea City Association Football Club (founded in **1912**) is a professional football club based in ...)*

Table 3.3: Classification of the reasoning necessary to answer questions in VIMQA, including English interpretations in *italics*. The linking entity is displayed in **bold orange**. Supporting details for answers are shown in blue. The answers themselves are emphasized in **bold green**. Words representing the reasoning type are marked in **purple**.

3.3 Benchmark Settings

3.3.1 Data Splits

To create VIMQA, we collected and annotated 10,047 valid examples. For evaluation, we followed the configuration of HotpotQA, dividing our dataset into training, development, and testing sets. Cross-validation was performed using the HotpotQA model as the baseline, with the results shown in Table 3.4. The model correctly answered 40% of the questions, which were marked as train-normal and used as part of the training set.

Fold	Answer		Sup Fact		Joint	
	EM	F1	EM	F1	EM	F1
1	31.3	36.1	13.4	43.8	5.5	17.7
2	31.7	36.5	25.4	59.7	9.9	23.7
3	31.0	37.0	21.8	55.6	8.36	22.65
4	36.6	42.0	13.6	42.3	5.4	18.9
5	32.5	37.7	28.5	62.6	10.7	25.2

Table 3.4: Result of 5-fold cross-validation on VIMQA

We collected and annotated 10,047 examples for VIMQA. Using the same configuration as HotpotQA [70] in English, we divided the dataset into training, development, and testing sets. Cross-validation was performed by the HotpotQA model (baseline) 5 times to select noteworthy samples. The results, presented in Table 3.4, showed the model answered 40% of questions correctly. This 40% was labeled as ”train-normal” and used as part of the training set. The other 60% of questions, which the model was unable to answer, were deemed complex and used to evaluate advanced and complex reasoning. These complex examples were divided into three subsets: train-hard, validation, and test, as shown in Table 3.5.

Name	Desc.	Usage	# Examples
train-normal	normal questions	train	4,018
train-hard	hard questions	train	4,023
dev	hard questions	validation	1003
test	hard questions	test	1003
Total			10,047

Table 3.5: VIMQA's data division

3.3.2 Benchmark settings

We have created two evaluation benchmark settings, based on the work of Yang et al. [70]. The benchmarks are named ”Gold Only” and ”Distractor” and both use the same test set samples with slight input differences.

The Gold Only setting measures a model's ability to perform multi-hop reasoning to answer a question and provide sentence-level supporting facts to explain its answer. In this setting, models receive two gold paragraphs and a question. Advanced multi-hop reasoning is required to answer.

The Distractor setting assesses a model's capability to identify the answer and supporting facts with the presence of distractions from other paragraphs. In this benchmark, models receive ten paragraphs (two gold and eight distractors) and must find the correct answer and supporting facts among them. To create this benchmark, we used the question as a query and selected eight summary paragraphs from Wikipedia with TF-IDF [7]. The two gold paragraphs containing the question and answer are combined with the eight distractors, resulting in ten paragraphs for each example in the distractor set, which are shuffled before use.

3.4 Experiments

3.4.1 Experimental Settings

Previous multi-hop QA methods were mostly developed for English. We have recreated leading multilingual QA models and tested them on the Vietnamese VIMQA dataset. These models have demonstrated success on English QA benchmarks such as SQuAD [50] and on Vietnamese benchmarks including UIT-ViQUAD [42].

Below are the specifics of our competitive baselines:

- BERT [16]: a widely used model in many NLP tasks. Our evaluation

Settings	**Methods**	**Answer EM**		**Answer F1**	
		Dev	Test	Dev	Test
Gold Only	mBERT	**56.63**	**55.03**	**71.27**	**70.50**
	XLM-RoBERTa$_{\text{Base}}$	47.35	43.76	62.70	59.38
	XLM-RoBERTa$_{\text{Large}}$	50.14	49.75	66.42	65.64
	InfoXLM$_{\text{Base}}$	50.54	49.05	67.68	65.76
	InfoXLM$_{\text{Large}}$	50.65	49.75	66.09	65.29
Distractor	BM25 + mBERT	**41.77**	**39.08**	**51.17**	**49.34**
	BM25 + XLM-RoBERTa$_{\text{Base}}$	29.31	29.11	40.04	39.47
	BM25 + XLM-RoBERTa $_{\text{Large}}$	32.20	32.30	42.33	43.80
	BM25 + InfoXLM$_{\text{Base}}$	36.19	34.39	47.59	45.82
	BM25 + InfoXLM$_{\text{Large}}$	31.40	31.10	43.24	42.53
	Human	**87.40**		**91.26**	

Table 3.6: Performance assessment of evaluated method on dev and test sets of VIMQA under two benchmark configurations

employs multilingual BERT (mBERT), which has been pre-trained on a large number of languages, with Vietnamese being one of them. Only the mBERT$_{\text{Base}}$ version is accessible for multilingual setup.

- XLM-RoBERTa [12]: this model delivers significant performance for various cross-lingual transfer tasks. In our experiments, we assess two versions: XLM-RoBERTa$_{\text{Base}}$ and XLM-RoBERTa$_{\text{Large}}$.

- InfoXLM [8]: an Information-Theoretic framework for cross-lingual language model sharing the same architecture as XLM-RoBERTa, with improved cross-lingual transfer ability. Our experiments evaluate two versions: InfoXLM$_{\text{Base}}$ and InfoXLM$_{\text{Large}}$.

For finding answers to Yes/No questions, we prepend two tokens indicating *Yes* and *No* to the context. This creates a context in which the answer span for Yes/No questions is present, enabling the model to find the answers.

Additionally, to demonstrate VIMQA's greater reasoning demands compared to other Vietnamese QA datasets, we compare VIMQA to UIT-ViQUAD [42]. To do this, we use results from [42] for XLM-RoBERTa and mBERT, run our own implementation of InfoXLM on UIT-ViQUAD, and use the results for comparison.

In accordance with the benchmark methodology outlined in Section 3.3, the performance of the models is evaluated in two scenarios of VIMQA (Gold Only and Distractor). The Distractor setting employs BM25 to select two out of ten paragraphs based on the question as the query, which are then passed to the QA model to get the answer. For the Gold Only scenario, the QA model alone is utilized to obtain the answer span from each sample.

Lastly, to assess the overall performance of the Multi-hop QA system, we replicated the baseline model presented by Yang et al. [70] and applied it to the VIMQA dataset, using three performance metrics for multi-hop QA: answer, supporting facts, and joint. We use sentences containing the answer span as the baseline to measure the supporting facts metric of the QA models mentioned above.

We use two metrics from Rajpurkar et al. [50] and Yang et al. [70] to evaluate the answer: exact match (EM) and F1. Additionally, we adopt two metrics from Yang et al. [70] to gauge the models' explainability: EM and F1 on the set of supporting facts compared to the gold set, and a combination of answer span and supporting fact evaluation referred to as joint metrics.

3.4.2 Human Performance

To measure human performance, we selected 500 random examples from the Distractor setting of the VIMQA development and test sets and assigned them to three Vietnamese-speaking researchers to obtain answers and supporting facts. We then compared the original gold annotations with the researchers' predictions using answers, supporting facts, and joint evaluation metrics. This serves as the human performance benchmark for the VIMQA dataset.

3.4.3 Results

The results of the evaluated models on the development and test sets of VIMQA, compared to human performance, are presented in Table 3.6. The data implies that VIMQA is a challenging dataset for current QA models, with the Distractor setting being more difficult than the Gold Only. Although mBERT performed best, it still falls significantly short of human performance.

The comparison of models' performance between VIMQA and UIT-ViQUAD is shown in Table 3.7. To make a fair comparison, the models are evaluated in the Gold Only setting of VIMQA, where only two gold paragraphs are provided. The results demonstrate that VIMQA is a more challenging dataset than UIT-ViQUAD, one of the most extensive Vietnamese span-extraction datasets. The results indicate that existing methods find VIMQA to be more challenging than UIT-ViQUAD.

Method	Split	VIMQA		UIT-ViQuAD	
		EM	F1	EM	F1
XLM-RoBERTa $_{\text{Base}}$	dev	47.35	62.70	63.87	81.90
	test	43.76	59.38	63.00	81.95
XLM-RoBERTa $_{\text{Large}}$	dev	50.14	66.42	69.18	87.14
	test	49.75	65.64	68.98	87.02
mBERT	dev	**56.63**	**71.27**	62.20	80.77
	test	**55.03**	**70.50**	59.28	80.00
InfoXLM $_{\text{Base}}$	dev	50.54	67.68	65.94	82.81
	test	49.05	65.76	64.36	82.39
InfoXLM $_{\text{Large}}$	dev	50.65	66.09	**72.52**	**88.85**
	test	49.75	65.29	**69.34**	**87.43**

Table 3.7: Evaluation of models' capabilities on VIMQA under the Gold Only setting and comparison with UIT-ViQUAD

Method	Answer		Sup Fact		Joint	
	EM	F1	EM	F1	EM	F1
Baseline	16.95	27.92	**25.12**	**53.42**	4.89	16.88
BM25 + InfoXLM $_{\text{Large}}$	31.10	42.53	19.34	31.45	**11.07**	21.94
BM25 + XLM-R$_{\text{Large}}$	32.30	43.80	20.64	32.86	10.97	**22.14**
BM25 + mBERT	**39.08**	**49.34**	18.04	31.33	7.87	18.30
Human	87.40	91.26	72.20	79.39	72.20	77.12

Table 3.8: Evaluating existing methods using three metrics on the Distractor test set of VIMQA

Finally, the comparison of the performance of the selected methods, baseline, and human performance is presented in Table 3.8 for the Distractor test set of VIMQA. The results indicate that while the selected models outperform the baseline, they lag significantly behind human performance across all three metric sets.

3.5 Summary

Our work introduces VIMQA, a multi-hop Vietnamese QA dataset, to promote the development of advanced reasoning QA models in Vietnamese. We present a pipeline for collecting generalized multi-hop QA examples and show its efficacy through a detailed analysis of VIMQA. Results from experiments demonstrate the difficulty and potential of VIMQA in both single and multi-hop QA, making it a valuable resource for both Vietnamese and cross-lingual QA models, especially in the area of reasoning and explainable answers with supporting facts.

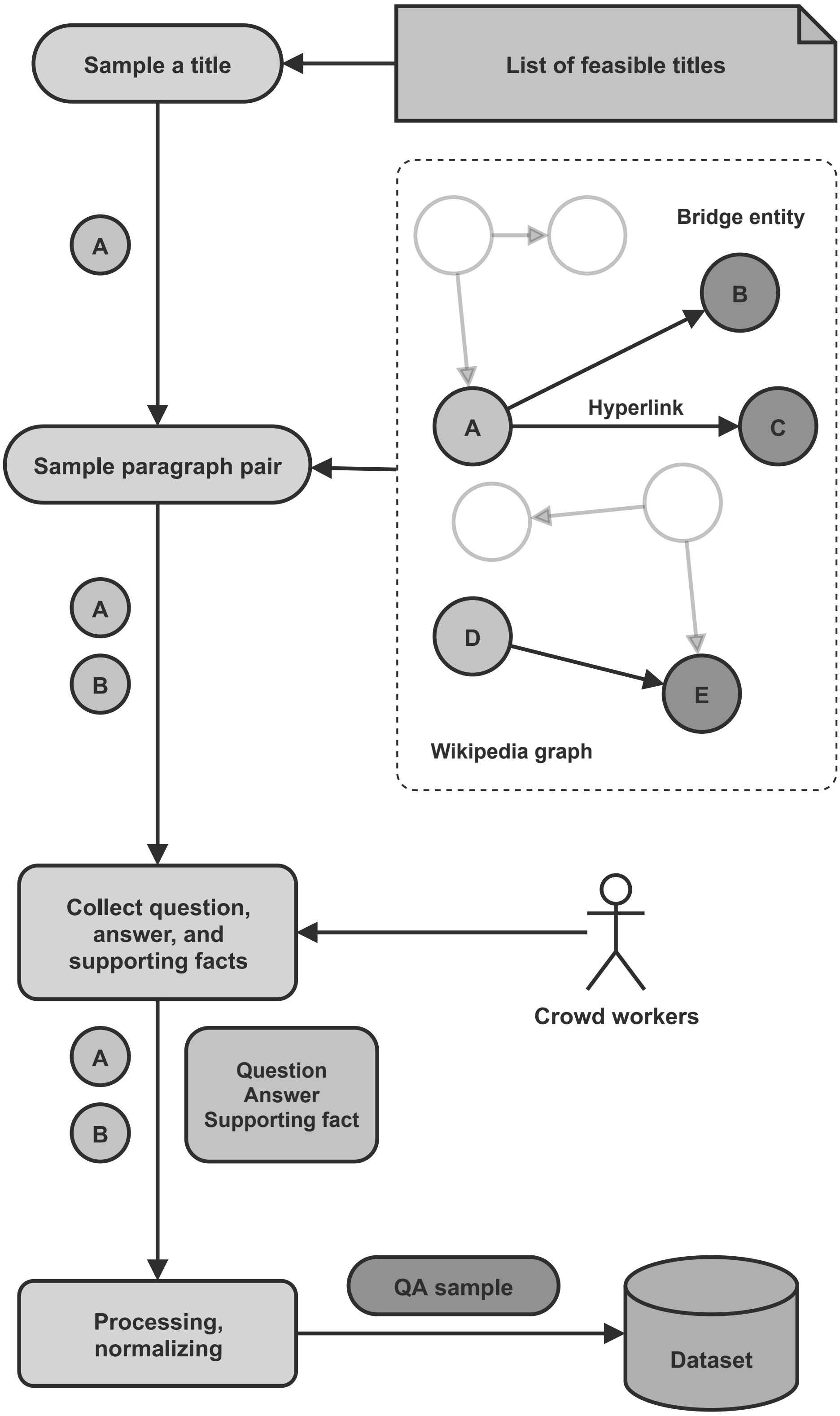
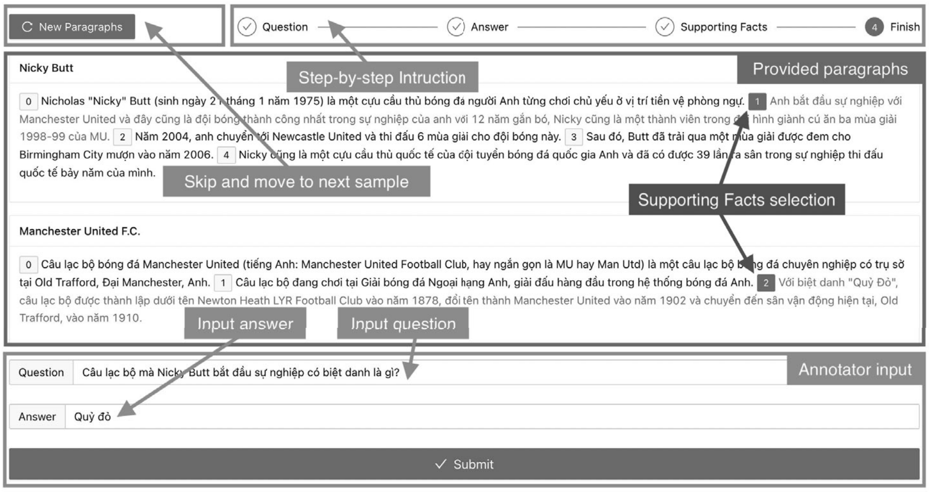


Figure 3.1: Overall data collecting pipeline of VIMQA



Nicky Butt

0 Nicholas "Nicky" Butt (sinh ngày 21 tháng 1 năm 1975) là một cựu cầu thủ bóng đá người Anh từng chơi chủ yếu ở vị trí tiền vệ phòng ngự. 1 Anh bắt đầu sự nghiệp với Manchester United và đây cũng là đội bóng thành công nhất trong sự nghiệp của anh với 12 năm gắn bó, Nicky cũng là một thành viên trong đội hình giành cú ăn ba mùa giải 1998-99 của MU. 2 Năm 2004, anh chuyển tới Newcastle United và thi đấu 6 mùa giải cho đội bóng này. 3 Sau đó, Butt đã trải qua một mùa giải được đem cho Birmingham City mượn vào năm 2006. 4 Nicky cũng là một cựu cầu thủ quốc tế của đội tuyển bóng đá quốc gia Anh và đã có được 39 lần ra sân trong sự nghiệp thi đấu quốc tế bảy năm của mình.

Manchester United F.C.

0 Câu lạc bộ bóng đá Manchester United (tiếng Anh: Manchester United Football Club, hay ngắn gọn là MU hay Man Utd) là một câu lạc bộ bóng đá chuyên nghiệp có trụ sở tại Old Trafford, Đại Manchester, Anh. 1 Câu lạc bộ đang chơi tại Giải bóng đá Ngoại hạng Anh, giải đấu hàng đầu trong hệ thống bóng đá Anh. 2 Với biệt danh "Quỷ Đỏ", câu lạc bộ được thành lập dưới tên Newton Heath LYR Football Club vào năm 1878, đổi tên thành Manchester United vào năm 1902 và chuyển đến sân vận động hiện tại, Old Trafford, vào năm 1910.

Question: Câu lạc bộ mà Nicky Butt bắt đầu sự nghiệp có biệt danh là gì?

Answer: Quỷ đỏ

Figure 3.2: User interface for annotators to input a sample

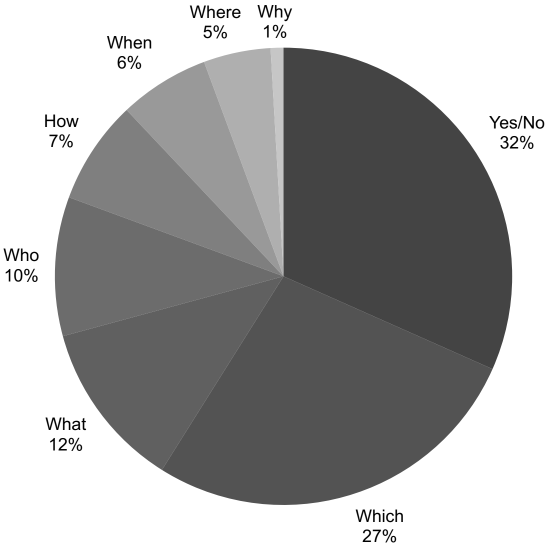


Figure 3.3: Percentage of question types in VIMQA

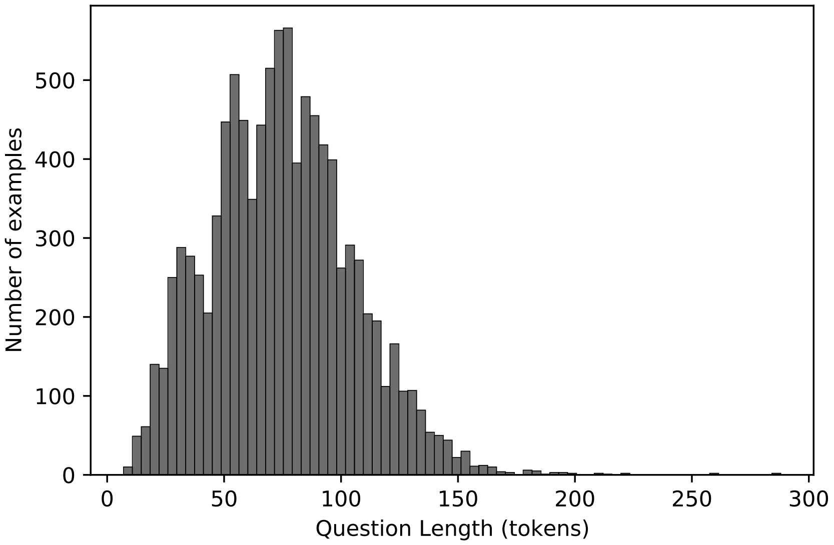


Figure 3.4: Distribution of question lengths in VIMQA

Chapter 4

Enhancing Reader performance by identifying relevant information

This chapter proposes a flexible Potential Sentence Classification model and pipeline to enhance the performance of current QA systems. Besides, our models are also ideal to be integrated and adapted into most popular QA systems, even in multilingual domains such as Vietnamese documents. Especially to deal with the massive documents in scientific domains, our method also proves its potential and effectiveness against the current competitive baselines.

4.1 The Proposed System

4.1.1 Overall

Our proposed pipeline comprises of two main stages. The initial step involves creating a reduced context using a classification model. The original context is divided into individual sentences and then processed through our proposed Potential Sentence Classification Model. A threshold adjustment process is applied for each context to select sentences classified as potential to form a new condensed context. In the following step, the new context and the question are inputted into the QA model to obtain the answer span. The overall process of the proposed pipeline is illustrated in Figure 4.1.

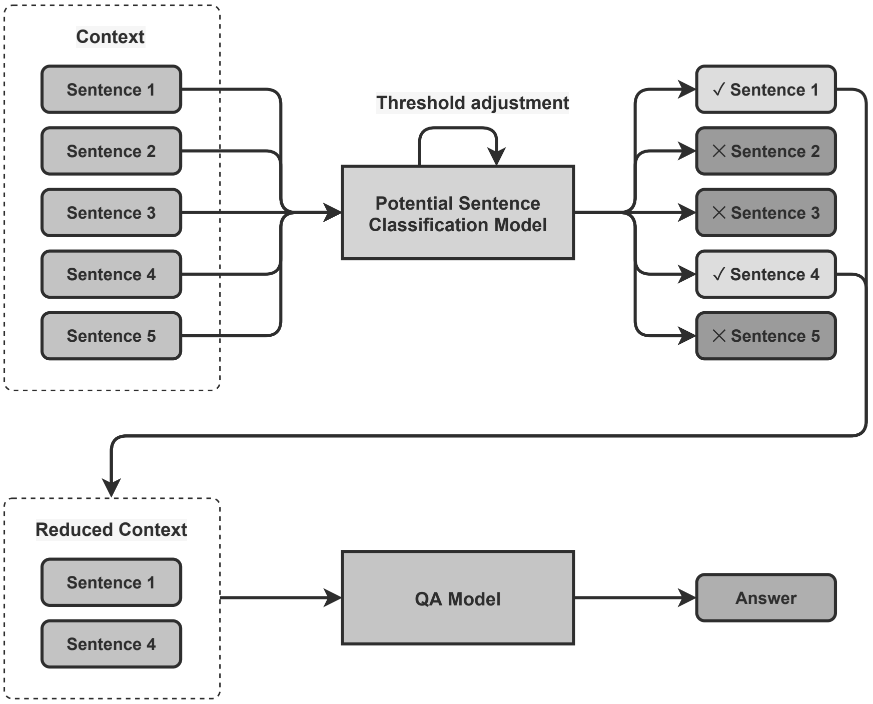


Figure 4.1: An overview of proposed system

4.1.2 Potential Sentence Classification Model

The Potential Sentence Classification Model (PSCM) is the core component of our pipeline. Its input is a pair of sentences: a question and a candidate sentence in the context. The objective of our PSCM is to identify whether a candidate sentence includes the answer to the question or not. Our PSCM is built using a transformer-based approach. In particular, we utilize RoBERTa[73], XLM-RoBERTa[11], and Sentence-BERT[52] depending on the dataset.

Data generation and model fine-tuning

We construct the PSCM by utilizing transfer learning with a pre-trained transformer-based model (RoBERTa) for the target QA dataset. We create the PSCM. To do this, we propose a method for generating the sentence-pair dataset for PSCM training from the QA resources. Particularly, the generation rule is as follows: For a context and question in the QA training set, if a sentence in the context is relevant to the question, the classification label for that sentence and question will be 1. Otherwise, it will be 0. We described the generated dataset for PSCM from SQuAD 2.0 in Section 4.2.

Threshold Adjustment

A fixed threshold can not work well for every question and context. Therefore, we propose a procedure to adjust the threshold for each context and question. Our constraint is that the length of the target context (reduced context) has to be in the range $(minLength, maxLength)$. The $minLength$ and $maxLength$ are hyperparameters and are determined based on the dataset and task. A binary-search technique is employed to find a suitable threshold that satisfies this constraint. Algorithm 1 describes in detail the method to determine the threshold for each context and question. Particularly, the number of sentences or tokens is decided by the threshold of $minLength$ and $maxLength$. The sentences are selected by the relevant score of sentence and question from $PSCM()$ and concatenated by $makeContext()$ to create the new concise context.

Algorithm 1 Threshold adjustment algorithm

```
Require: $minLength, maxLength, question, context$
Ensure: $minLength < maxLength$
  $minThreshold \leftarrow 0$
  $maxThreshold \leftarrow 1$
  while $minThreshold < maxThreshold$ do
      $threshold \leftarrow (minThreshold + maxThreshold)/2$
      $sentences \leftarrow sentenceSegment(context)$
      $potentialSentences \leftarrow PSCM(sentences, question)$
      $reducedContext \leftarrow makeContext(potentialSentences)$
      if length of $reducedContext <= minLength$ then
          $maxThreshold \leftarrow threshold$
      else if length of $reducedContext >= maxLength$ then
          $minThreshold \leftarrow threshold$
      else
          return $threshold$
      end if
  end while
```

4.1.3 Answer Extraction

The second step of the pipeline utilizes a QA model to identify the specific section of the context that contains the answer. This step is independent of the first step and can be used with any QA model. We conduct experiments using various state-of-the-art transformer-based QA models, following the implementation of Transformers[63]. This involves adding a linear layer, known as a span classification head, on top of the hidden-states output to determine the starting and ending positions of the answer. The final answer is calculated by choosing the valid pair of start and end logit with the highest sum of the two values. The corresponding tokens with the selected start and end logit values are the answer start and end positions.

4.2 Experiments and Results

4.2.1 Dataset

To demonstrate the effectiveness and versatility of our model, we train and evaluate it using three distinct datasets and various commonly used QA models. The specifics of the datasets are outlined below:

- **Qasper**[13] is a QA dataset on Natural Language Processing (NLP) papers where questions and answers are provided by NLP practitioners. The context for each question is an entire scientific research paper whose size is massive compared to other QA datasets. Qasper is demonstrated to pose a challenge for current state-of-the-art models.

- **UIT-ViQuAD**[40], a manual crowd-sourced span-extraction dataset for Vietnamese machine reading comprehension (MRC) systems, was created using Vietnamese Wikipedia and contains 23k question-answer pairs from 5k passages. It is among the few large-scale Wikipedia-based datasets for evaluating Vietnamese QA systems.

- **SQuAD 2.0**[48] is a crowd-sourced reading comprehension dataset of questions about Wikipedia articles, where answers are text spans from the corresponding passages. It combines the 100k questions from SQuAD1.1 with 50k unanswerable questions crafted to appear similar to answerable ones.

The data analyses of three datasets are shown in Table 4.1. There are three main points in our comparison. Firstly, it is valuable to prove the effectiveness of our model in the general domain via SQuAD 2.0 against the most popular QA systems. Secondly, we also emphasize the potential of our pipeline in multilingual adaption via the Vietnamese UIT-ViQuAD dataset. It reveals the novelty of our model in this language, where we propose the flexible threshold in context filtering. Finally, we also point out the promising results of our models in scientific documents whose contexts are highly huge in length.

Dataset	**Detail**	**All**	**Train**	**Dev**	**Test**
Qasper	#questions	5,049	2,593	1,005	1,451
SQuAD 2.0	#articles	505	442	35	28
	#questions	151,051	130,319	11,873	8,862
UIT-ViQuAD	#articles	174	138	18	18
	#passages	5,109	4,101	515	493
	#questions	23,074	18,579	2,285	2,210
	Average passage length	153.4	153.9	147.9	155.0
	Average question length	12.2	12.2	11.9	12.2
	Average answer length	8.2	8.1	8.4	8.9
	Vocabulary size (words)	41,773	36,174	9,184	9,792

Table 4.1: The detailed analysis of the datasets in the experiments.

As we mentioned above, we also propose a process to generate the dataset for the PSCM module. We apply the proposed method to generate the dataset for training the PSCM module from UIT-ViQuAD and SQuAD 2.0. The detail of our extracted dataset is presented in Table 4.2. For Qasper dataset, a context for a question is an entire paper, and the negative sentences (sentences that do not contain the answer) are dominant compared to positive sentences (sentences containing the answer). As a result, the process of generating training data for PSCM for Qasper is not trivial and requires more research. Therefore, we do not fine-tune the PSCM module in our experiments on Qasper dataset.

Source Dataset	**Detail**	**All**	**Train**	**Validation**
UIT-ViQuAD	Number of samples	116,038	102,972	13,066
	Number of label 1	20,865	18,579	2,286
	Number of label 0	95,173	84,393	10,780
SQuAD	Number of samples	718,295	655,404	62,891
	Number of label 1	106,113	98,439	7,674
	Number of label 0	612,182	556,965	55,217

Table 4.2: Overview of the generated datasets for PSCM.

4.2.2 Models

Models for Qasper dataset

For the PSCM module, we employ SBERT [52] and the pre-trained SBERT model **all-mpnet-base-v2** to get the embedding of sentences. The cosine-similarity score between the sentence in the context and the question is calculated and compared with the threshold to choose the potential sentences. For answer extraction, we employ the implementation of **Qasper-LED** model proposed by [13], which is based on Longformer-Encoder-Decoder (LED)[3]. We conduct experiments to evaluate the improvement of Qasper-LED when applying our pipeline.

Models for UIT-ViQuAD dataset

For the PSCM, the multilingual model XLM-RoBERTa$_{\text{Large}}$ with a sequence regression head on top is used. We utilize the implementation of XLM-RobertaForSequenceClassification from Wolf et al. [63]. For answer extraction, the following state-of-the-art multilingual QA models are applied.

- **Multilingual BERT (mBERT)** [15]: The multilingual version of BERT, one of the most popular models in many NLP tasks. mBERT is pre-trained in 104 languages, including Vietnamese.
- **XLM-RoBERTa** [11]: A state-of-the-art multilingual model that has significant performance for a variety of cross-lingual transfer tasks. In our experiments, we evaluate two versions of this model, XLM-RoBERTa$_{\text{Base}}$ and XLM-RoBERTa$_{\text{Large}}$.

We also conducted experiments to compare our method with the ViReader system, one of the state-of-the-art MRC systems trained and evaluated in Vietnamese. For comparison, we take the ViReader API and training source codes from the original paper and reproduce the result in our experiment environment.

Models for SQuAD 2.0 dataset

For the PSCM, we use RoBERTa$_{\text{Large}}$ with a sequence regression head on top is used. We utilize the implementation of RobertaForSequenceClassification from Wolf et al. [63]. For answer extraction, the following methods are used.

- **RoBERTa**: The model was proposed by Zhuang et al. [73]. It improves BERT by adjusting key hyperparameters, removing the next-sentence pretraining objective, and training with much larger mini-batches and learning rates. For the QA task, RoBERTa achieves remarkable results in SQuAD 2.0 dataset. We conduct experiments on two versions of RoBERTa (Base and Large).

- **ELECTRA**: The model was proposed by Clark et al[9]. It employs a new pretraining approach that trains two transformer models: the generator and the discriminator. ELECTRA achieves noticeable results on QA benchmarks like SQuAD and HotpotQA. We conduct experiments on the Base version of ELECTRA
- **BERT**: The model was proposed by Devlin et al.[15] as a bidirectional transformer pre-trained using a mixture of masked language modeling objective and next sentence prediction. We conduct experiments on the Base-Case version BERT.

4.2.3 Experimental Results

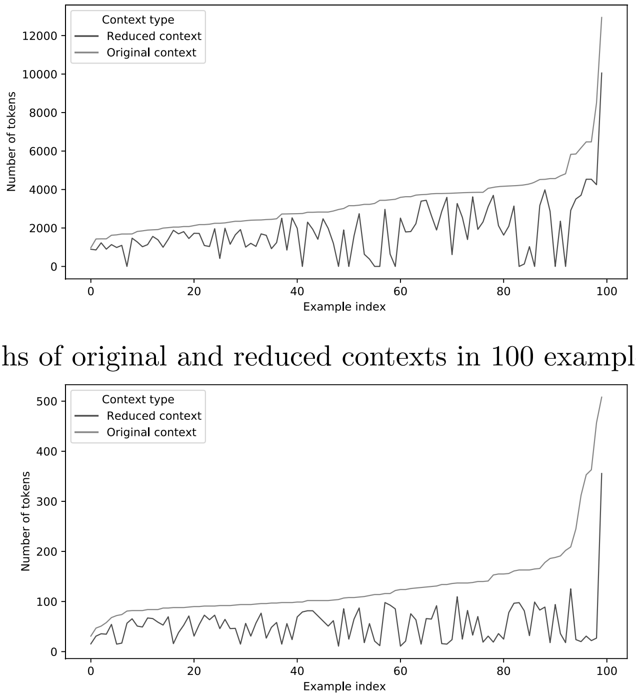
We first conduct an experiment to evaluate the performance of the PSCM module. The PSCM module is trained using the train set and evaluated using the validation set in the dataset described above. Table 4.3 shows the result of the PSCM module evaluation.

Source Dataset	Accuracy (%)	AUPRC (%)	AUROC (%)	Precision (%)	Recall (%)	F1 (%)
SQuAD 2.0	94.36	81.18	94.19	84.47	65.90	74.04
UIT-ViQuAD	91.47	87.23	94.46	89.36	66.49	76.25

Table 4.3: Result of the PSCM module evaluation on the generated dataset

To visualize how much our method reduced the context in the SQuAD dataset. We randomly sample several examples from the SQuAD dataset and plot the lengths of the original contexts and the contexts reduced by our method. We sort the examples based on the original context lengths to make them easy to interpret. Figure 4.2 visualizes the amount of distracting information removed by employing our method. The space between the "Original context" line and the "Reduced context" line denotes the portion of the context reduced by our method.

After proving the strength of our proposed module to reduce the context, we also present the effectiveness of our pipeline in general QA systems. Firstly, Table 4.4 shows the overall results of Qasper-LED model on the Qasper test set when applying our pipeline to improve the performance. The



(a) Lengths of original and reduced contexts in 100 examples in Qasper

(b) Lengths of original and reduced contexts in 100 examples in SQuAD 2.0

Figure 4.2: Comparing the number of tokens in the original and reduced contexts

result is shown with the performance breakdown on the different answer types. The result reveals that our pipeline successfully enhances the overall performance of Qasper-LED, especially in Extractive and Yes/No questions. The other types of questions, including Abstractive and Unanswerable are not suitable for context reduction. The reason for this phenomenon comes from its requirement of the general relationship in content to find out the abstract answer as well as conflict between input documents and questions.

Method	**Extractive**	**Abstractive**	**Yes/No**	**Unanswerable**	**Overall**
Qasper-LED (Single Model)	27.53	14.78	60.87	49.49	30.58
Qasper-LED (Our Method)	29.69	14.31	65.78	41.67	**31.30**

Table 4.4: Result on Qasper dataset of single model and our method

Secondly, it is valuable to digest the experimental results in the Vietnamese QA dataset. Table 4.5 compares the performance of state-of-the-art methods in multilingual QA models on the ViQuAD dataset when applying our pipeline. The result shows that our method achieves better F1 scores in all three evaluated models. Besides, we also compare our method against the

SOTA QA system in UIT-ViQuAD named ViReader. Table 4.6 presents the details of our comparison. In particular, we use the version of our method applying on XLM-RoBERTa$_{\text{Large}}$, which has the highest performance in our experiment on ViQuAD. It is easily noticed that the performance of ViReader depends on the number of sentences (K) which is pre-defined and fixed for all samples in the retrieval module. The result indicates that our method outperforms the ViReader on the ViQuAD dataset with a flexible threshold learned by our Algorithm 1.

Model	Single Model		Our Method	
	EM	F1	EM	F1
XLM-RoBERTa $_{\text{Large}}$	**73.59**	88.74	73.27	**89.06**
XLM-RoBERTa $_{\text{Base}}$	63.72	81.54	**64.08**	**82.56**
mBERT	58.83	77.72	**59.82**	**78.98**

Table 4.5: Result on UIT-ViQuAD dataset of single model and our method

K-sentences retrieved	ViReader		Our Method (with XLM-RoBERTa$_{\text{Large}}$)	
	EM	F1	EM	F1
1	55.20	67.94	73.27	**89.06**
2	63.90	78.92		
3	69.29	84.57		
4	71.37	86.83		
5	72.19	87.70		
6	73.41	88.52		
7	73.46	88.50		
8	73.55	88.60		
9	73.59	88.74		
10	73.59	88.80		

Table 4.6: Compares our method(applying on XLM-RoBERTa) and the ViReader. The ViReader depends on the numbers of sentences (K) in the retrieval step

Finally, table 4.7 compares the result of these models on the general domain via SQuAD 2.0 dataset when using a single model and applying our pipeline. We use the metric Exact Match (EM), and F1 score (F1) proposed by Rajpurkar et al.[51] for evaluation. The result shows that our method produces better results when applied to any of the four models. In the RoBERTa Large model, our method successfully increases the EM to 82.69% (almost 1.0 point improvement) and the F1 score to 85.78% (over 1.0 point improvement).

Model	Single Model		Our Method	
	EM	F1	EM	F1
RoBERTa $_{\text{Large}}$	81.75	84.57	**82.69**	**85.78**
RoBERTa $_{\text{Base}}$	76.48	79.48	**78.94**	**82.02**
ELECTRA $_{\text{Base}}$	64.64	69.15	**65.01**	**69.46**
BERT	71.47	74.98	**71.78**	**75.25**

Table 4.7: Result on SQuAD dataset of single model and our method

4.3 Discussion

To provide a better understanding of the improvements and the limits of our proposed methods for the sentence retrieval module, we discuss two examples in this section.

Question: Hơn phân nửa số người Đức nhưng không có quyền công dân Đức là sống ở đâu? (*Where do more than half of Germans without German citizenship live?*)

Answer: miền tây của liên bang và hầu hết là tại các khu vực đô thị (*western part of the federation and mostly in urban areas*)

Our retrieved passage: Có khoảng 5 triệu người có quốc tịch Đức cư trú tại nước ngoài (2012). Năm 2014, có khoảng bảy triệu người trong số 81 triệu cư dân Đức không có quyền công dân Đức. Sáu mươi chín phần trăm trong số đó sống tại miền tây của liên bang và hầu hết là tại các khu vực đô thị. Đức xếp hạng bảy trong EU và thứ 37 toàn cầu về tỷ lệ người nhập cư so với tổng dân số. Từ năm 1987, có khoảng 3 triệu người dân tộc Đức, hầu hết từ các quốc gia Khối phía Đông, đã thực hiện quyền trở về của mình và di cư đến Đức. (*There are about 5 million German nationals residing abroad (2012). In 2014, about seven million of Germany's 81 million residents did not have German citizenship. Sixty-nine percent of them live in the western part of the federation and most are in urban areas. Germany ranks seventh in the EU and 37th globally in terms of immigration to total population. Since 1987, about 3 million ethnic Germans, mostly from Eastern Bloc countries, have exercised their right to return and emigrate to Germany*)
Our answer: miền tây của liên bang và hầu hết là tại các khu vực đô thị (*western part of the federation and mostly in urban areas*)
Score: $EM = 1$, $F1 = 1$

The STR retrieved passage: Có khoảng 5 triệu người có quốc tịch Đức cư trú tại nước ngoài (2012). Năm 2014, có khoảng bảy triệu người trong số 81 triệu cư dân Đức không có quyền công dân Đức. Năm 2015, Đức là quốc gia có số lượng di dân quốc tế cao thứ hai thế giới, với khoảng 5% hay 12 triệu người. Đức xếp hạng bảy trong EU và thứ 37 toàn cầu về tỷ lệ người nhập cư so với tổng dân số. Từ năm 1987, có khoảng 3 triệu người dân tộc Đức, hầu hết từ các quốc gia Khối phía Đông, đã thực hiện quyền trở về của mình và di cư đến Đức. (*There are about 5 million German nationals residing abroad (2012). In 2014, about seven million of Germany's 81 million residents did not have German citizenship. In 2015, Germany was the country with the second highest number of international migrants in the world, with about 5 % or 12 million people. Germany ranks seventh in the EU and 37th globally in terms of immigration to total population. Since 1987, about 3 million ethnic Germans, mostly from Eastern Bloc countries, have exercised their right to return and emigrate to Germany.*)
The reproduced ViReader's answer: nước ngoài (*foreign country*)
Score: $EM = 0$, $F1 = 0$

Table 4.9: Compares the context reduced using our pipeline and using ViReader retrieval module. The correct answer is highlight in red

Table 4.8 shows the first example where distracting information affects the model decision in SQuAD 2.0. The highlighted text is the exact answer to the question in this example. With the reduced context, RoBERTa model can answer with F1 score = 1 and Exact Match = 1. With the original context, the same RoBERTa model can not identify the answer span and arrive at the empty string answer, with the F1 score = 0 and Exact Match = 0. This example indicates that our pipeline selects the sentences that contain the answer span and successfully removes distracting information. In addition, it also shows that too many distracting details can hurt the QA model's performance noticeably.

Table 4.9 shows the contexts reduced using our pipeline and using ViReader retrieval module. In this example, the highlighted text is the exact answer to the question. Our system has the correct answer with F1 score = 1 and Exact Match = 1 while the ViReader's answer has F1 score = 0 and Exact Match = 0. It is clear that our system successfully retrieves the sentence that contains the answer span. This enables the answer extracting model to find the correct answer. In contrast, the ViReader retrieval module cannot retrieve the sentence with the answer span. This leads to poor results in the answer extraction module.

4.4 Summary

In this work, we propose a novel model-agnostic pipeline to remove distracting information from the contexts of the span-extraction QA task. The proposed method successfully improves existing QA models' performance through the Potential Sentence Classification Model (PSCM) and the Threshold Adjustment algorithm. In addition, we also propose a delegate process to extract the training dataset for PSCM from the original QA resources. The experimental results show that our method remarkably enhances existing QA models and can be applied to a wide range of models and datasets. Our pipeline is especially useful in QA in scientific documents, which have massive and complex contexts. In addition, using the state-of-the-art multilingual model in QA, our pipeline achieve state-of-the-art performance on ViQuAD dataset in Vietnamese. Our detailed discussion reveals how distracting information affects the model's decision and the necessity of our method.

Question: Who was the duke in the battle of Hastings?
Answer: William the Conqueror
The Reduced Context: Norman adventurers founded the Kingdom of Sicily under Roger II after conquering southern Italy on the Saracens and Byzantines, and an expedition on behalf of their duke, William the Conqueror, led to the Norman conquest of England at the Battle of Hastings in 1066. **RoBERTa answer**: “William the Conqueror” **Score**: $EM = 1$, $F1 = 1$
The Original Context: The Norman dynasty had a major political, cultural and military impact on medieval Europe and even the Near East. The Normans were famed for their martial spirit and eventually for their Christian piety, becoming exponents of the Catholic orthodoxy into which they assimilated. They adopted the Gallo-Romance language of the Frankish land they settled, their dialect becoming known as Norman, Normaund or Norman French, an important literary language. The Duchy of Normandy, which they formed by treaty with the French crown, was a great fief of medieval France, and under Richard I of Normandy was forged into a cohesive and formidable principality in feudal tenure. The Normans are noted both for their culture, such as their unique Romanesque architecture and musical traditions, and for their significant military accomplishments and innovations. Norman adventurers founded the Kingdom of Sicily under Roger II after conquering southern Italy on the Saracens and Byzantines, and an expedition on behalf of their duke, William the Conqueror, led to the Norman conquest of England at the Battle of Hastings in 1066. Norman cultural and military influence spread from these new European centres to the Crusader states of the Near East, where their prince Bohemond I founded the Principality of Antioch in the Levant, to Scotland and Wales in Great Britain, to Ireland, and to the coasts of north Africa and the Canary Islands. **RoBERTa answer**: “” (empty string) **Score**: $EM = 0$, $F1 = 0$

Table 4.8: Example in SQuAD 2.0 where distracting information affects the model decision. The correct answer is highlight in red

Chapter 5

ViWiQA: Efficient End-to-end Vietnamese Wikipedia-based Open-domain Question-Answering Systems for Single-hop and Multi-hop Questions

To address the challenges in Vietnamese Open-domain QA, this chapter proposes new state-of-the-art multilingual retriever methods for single-hop and multi-hop Open-domain QA that can be efficiently trained with low resources. Using these retrievers, we proposed **ViWiQA**, a set of efficient end-to-end Vietnamese Open-domain QA systems taking Wikipedia as their knowledge base. ViWiQA consists of **ViWiQA-Single** and **ViWiQA-Multi** systems, for single and multi-hop QA, respectively. Our code, models, Lucene-BM25 and DPR indexing, and the Vietnamese Wikipedia hyperlink graph are accessible to the public to support other research about Vietnamese QA.

5.1 Method

5.1.1 Problem formulation

The Vietnamese Open-domain QA problem can be explained as follows. Given a question q in the Vietnamese natural language, a QA system must answer the question using a knowledge base C. The knowledge base C contains c passages, denoted as $P_1, P_2, ..., P_c$, in which passage P_i consists of tokens $p_i^{(1)}, p_i^{(2)}, ..., p_i^{(l)}$ with l is the length of the passage. For single-hop questions, the task is to retrieve one passage P_i among c candidate passages and use P_i to obtain the answer. For multi-hop questions, the task is to retrieve a pair of passage P_i, P_j among c candidate passages and connect P_i and P_j to get the answer to the question q.

5.1.2 Vietnamese Wikipedia Pre-processing

This section describes building the Vietnamese Wikipedia knowledge base for Open-domain QA and the Wikipedia hyperlink graph. The Vietnamese Wikipedia dump from January 20, 2022, is used as the source document. Inspired by the approach of Chen et al. [6], WikiExtractor was employed to get the text-only portion of the Wikipedia dump, removing semi-structured data like lists and tables. Upon obtaining the texts of all articles, a sliding window technique with window size $W = 100$ and stride $S = 50$ is employed to separate the articles into overlapping text chunks, each of which contains W words, following the work of Wang et al. [61]. These text blocks are considered passages and used as basic units for retrieval. This process results in 3,885,030 passages.

The Wikipedia hyperlink graph is extracted along with the Vietnamese Wikipedia passages. The Wikipedia hyperlink graph is a directed graph where each node is a Wikipedia article, and each edge (u, v) indicates there exists a hyperlink from the article u to article v. WikiExtractor is employed to get the text portion of the Wikipedia articles, preserving the HTML tags for hyperlinks. A regular expression is then applied to extract the set of linked titles from the HTML tags. For each article u, the set of linked titles V is extracted. For each linked title v_i in V, if there exists an article with the same title v_i after normalization, the edge (u, v_i) is added to the graph.

5.1.3 Retriever for Single-hop QA

We propose a retriever method for Vietnamese single-hop Open-domain QA, namely *ViWiQA-Single Retriever*. Figure 5.1 shows the process of the proposed method. Given a question, we first use Lucene-BM25 [33] to retrieve top m passages from the Wikipedia knowledge base. The value m must be sufficiently small to apply the Cross-encoder and Reader models in the next step. The question is then paired with each retrieved passage to create the question-passage pairs. The pairs are fed to the Cross-Encoder model to obtain the relevance scores of m passages concerning the question. At the

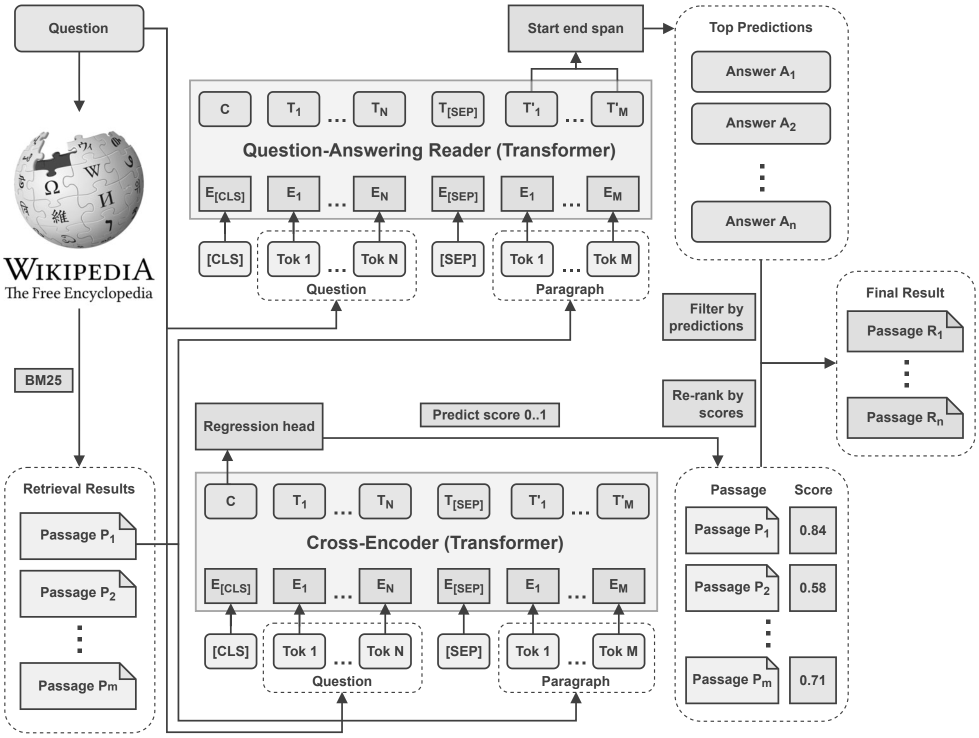


Figure 5.1: Overall of the proposed single-hop retriever component of ViWiQA-Single

same time, the pairs are also fed to the Reader model to extract the answer spans for each passage. Only top n answer span predictions with the highest confidence score are kept. Finally, the obtained relevance score and answer span predictions are used to rerank the passages. Passages are sorted using the relevance score, and only the passages that contain at least one of the n answer spans are selected. When checking if a passage contains an answer span, both the passage and answer span are normalized by converting to lowercase and removing punctuation. This process results in n reranked passages. The detailed architectures of the Cross-Encoder and Reader models are described in section 5.1.5 and 5.1.6, respectively.

5.1.4 Retriever for Multi-hop QA

We propose *ViWiQA-Multi Retriever*, a retriever method for Vietnamese multi-hop Open-domain QA. The retriever aims to retrieve the pair of passages that can be used to perform multi-hop reasoning and find the answer to a given question. Figure 5.2 shows the process of the proposed method.

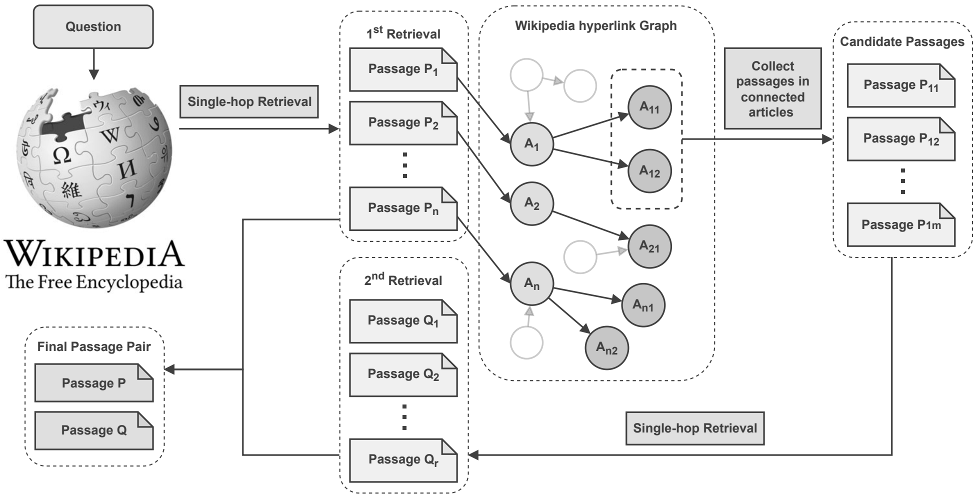


Figure 5.2: Overall of the proposed multi-hop retriever component of ViWiQA-Multi

Given a question, the same process of single-hop retriever described in Section 5.1.3 is carried out to retrieve top n passages $P_1, P_2, ..., P_n$. The passage with the highest rank (P_1) is chosen as the first passage in the passage pair. Let A_1 be the article containing the passage P_1; the c_1 articles connected to A_1 via hyperlinks $\{A_{11}, A_{12}, ..., A_{1c_1}\}$ are retrieved using the Wikipedia hyperlink graph. The passages of the connected articles are then collected and considered as candidates for the second passage in the pair. Using the candidate passages and the given question, the process of single-hop retriever is once again carried out to retrieve r passages $Q_1, Q_2, ..., Q_r$. The passage with the highest rank (Q_1) is chosen as the second passage for the passage pair. The final passage pair (P_1, Q_1) is then obtained.

In our experiments, only the highest-rank passage (P_1) is considered the first passage in the pair. As a result, only c_1 articles connected to P_1 in the hyperlink graph $\{A_{11}, A_{12}, ..., A_{1c_1}\}$ are used to get candidate passages for the second passage. However, it is also possible to consider lower-rank passages $P_2, ..., P_n$ and their corresponding connected articles in the hyperlink graph as shown below.

$$P_2 \rightarrow \{A_{21}, A_{22}, ..., A_{2c_2}\}$$
$$P_3 \rightarrow \{A_{31}, A_{32}, ..., A_{3c_3}\}$$
$$...$$
$$P_n \rightarrow \{A_{n1}, A_{n2}, ..., A_{nc_n}\}$$

A strategy to choose the final passage pair among the candidate pairs will then be needed. We leave this to future work.

5.1.5 Cross-Encoder model

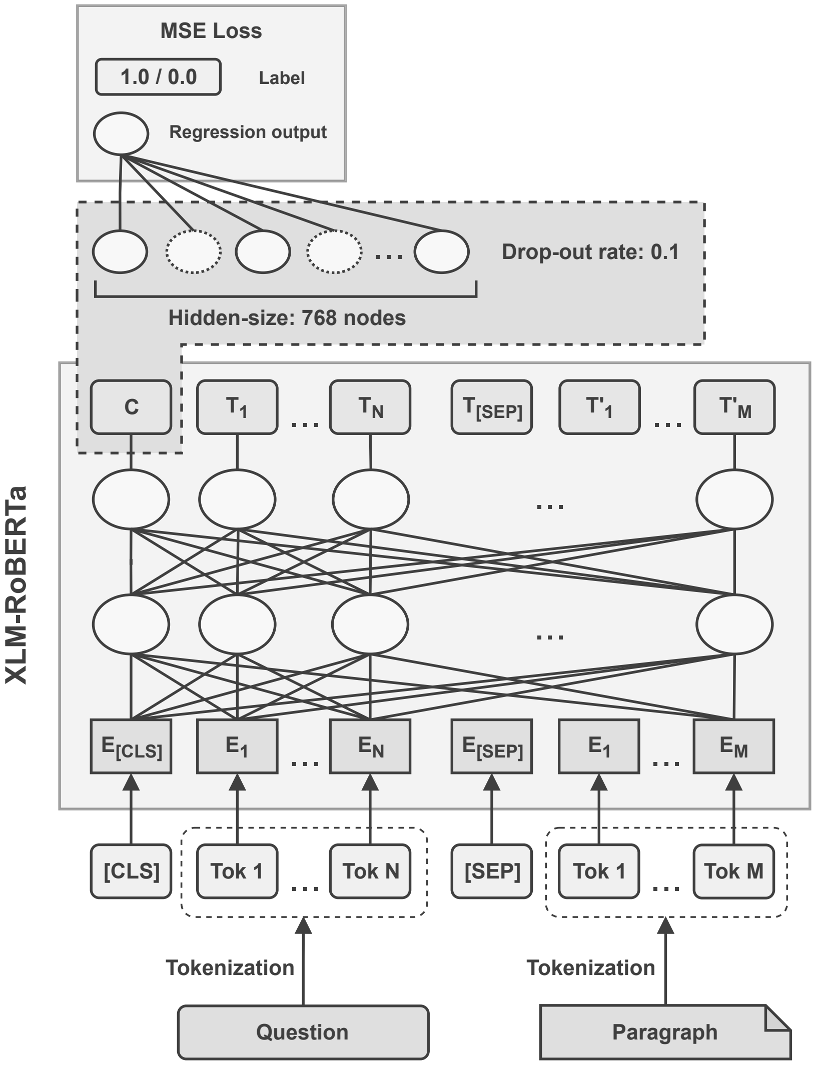


Figure 5.3: Detailed architecture of Cross-Encoder model for predicting paragraph relevance scores

To predict the relevance score for the question-passage pairs, the Cross-Encoder architecture [14] with Transformer [57] is employed, following the work of Nogueira and Cho [44]. Figure 5.3 shows the detailed architecture of the Cross-Encoder model. As the inputs are in Vietnamese, the pre-trained model XLM-RoBERTa [10] is used as our Transformer model. The input question and passage are concatenated and separated by a *[SEP]* token. The classification token *[CLS]* is added to the head of the sequence, representing the sentence-level classification. Following the work of Devlin et al. [14], we add a sequence regression head on top of the Transformer pooled output. The regression head outputs a score from 0 to 1, indicating the relevance of the passage with respect to the given question. We use the default configurations of the XLM-RoBERTa model where the last layer hidden-state of the classification token has the size of *768* and the drop-out rate of *0.1.* The sequence regression head has the input size *768* and the output size *1*, where the output indicates the regression output. An *Mean Square Error* (MSE) loss is calculated using the regression outputs and the labels. The label is *0.0* for negative examples and *1.0* for positive examples. The calculated MSE loss is used as the training objective to train the model. The MSE formula is given as follows.

$$MSE = \frac{1}{m}\sum_{i}(\widehat{y} - y)^2$$

where m is the number of example inputs, $\widehat{y}$ is the prediction of the model, and y is the regression target (label).

5.1.6 Question-Answering Reader model

For the Reader model, we employ XLM-RoBERTa [10] with a span classification head behind the final hidden-states to calculate the logits of a span being the start or end of the answer. Figure 5.4 describes the detailed architecture of our Reader. We use the default configurations of the large version of the XLM-RoBERTa model with 24 layers, and the size of the hidden layers is 1024. The two vectors $S \in \mathbb{R}^H$ and $E \in \mathbb{R}^H$ are introduced for calculating the start/end logits. The answer-start probability of a word i is calculated using the dot product of T'_i and S and is formulated as follows.

$$Pstart_i = \frac{e^{S.T'_i}}{\sum_j e^{S.T'_j}}$$

A similar formula is applied for the end of the span.

$$Pend_i = \frac{e^{E.T'_i}}{\sum_j e^{E.T'_j}}$$

A candidate answer spanning from i to j $(i \leq j)$ has the score computed as $S.T'_i + E.T'_j$. The span with the best score is selected to be the predicted answer.

5.1.7 Training

The Cross-Encoder and the QA Reader in ViWiQA require training to adapt to the Vietnamese QA datasets. Negative and positive examples first need to be sampled to train the Cross-Encoder. Question-passage pairs where the passage holds the gold answer are considered positive examples, while the pairs with the passage irrelevant to the question are considered negative examples. We follow one negative sampling approach proposed by [25] that uses top passages retrieved by Lucene-BM25, which have many matched tokens in the question but do not hold the answer. For each question and the gold answer in the dataset, Lucene-BM25 is used to retrieve top n passages and traverse through n passages from the highest-ranked to the lowest-ranked passage. A passage containing the gold answer is marked as a positive example. Otherwise, it is marked as a negative example. We continue the process until the ratio between the negative and positive examples of the given question exceeds a predefined value r. In our experiments, we use $n = 100$ and $r = 7$. We train the Cross-Encoder on the sampled training data. The number of epochs trained is 3; The initial learning rate is 1e-5; The batch size is 32.

The QA Reader of ViWiQA is trained using the annotated answer span start/end positions from the QA dataset. The number of epochs trained is 5; The initial learning rate is 1e-5; The batch size is 16. We observed that the reader model converged at around epoch 2.5.

5.1.8 End-to-end QA System

There are two main modules in the end-to-end QA system: Retriever and Reader. Given a question, the Retriever retrieves the most relevant passage from the knowledge base, and the Reader reads the relevant passage to find the answer. We proposed two separate QA systems for single-hop and multi-hop QA, namely *ViWiQA-Single* and *ViWiQA-Multi*, respectively. The systems employed the corresponding retrievers shown in Section 5.1.3 and 5.1.4.

We use the same Reader component, which has the architecture and training process shown in Section 5.1.3 and 5.1.7, for both QA systems. For ViWiQA-Single, the question and the passage retrieved using the single-hop retriever are used as the input for the Reader. For ViWiQA-Multi, the question and the passage pair retrieved using the multi-hop retriever are used as the input for the Reader. Upon being fed to the Reader, the two passages in the pair are concatenated to form a single passage. Because the Vietnamese multi-hop QA dataset VIMQA [29] has a similar format as HotpotQA [69] and contains Yes/No questions, a Yes/No tag in Vietnamese (”đúng/không”) is inserted at the beginning of the passage to enable the Reader to extract Yes/No answers.

5.1.9 Wikipedia-Entity-Resolution and Model Ensemble

In addition to ViWiQA, we also propose a simple approach to ER in QA and develop an entity-level end-to-end QA system (ViWiQA-ER) that accommodates the requirements of the Zalo AI Challenge 2022. In the Zalo AI Challenge contest, the End-to-end QA task over Wikipedia requires the output answer to be a Wikipedia entity, a specific date, or a number. Moreover, we propose an ensemble method for multiple retrievers and readers to boost the performance of the QA system. Figure 5.5 shows the overall end-to-end QA system with Wikipedia entity resolution.

For questions where the answers are entities, the goal is to convert the plain text answer from the QA model to the corresponding Wikipedia entity. We employ the *redirect pages* metadata from Wikipedia for entity resolution. In Wikipedia, whenever a user accesses a *redirect page*, it will redirect the user to another Wikipedia page that refers to the same entity. For example, whenever the Wikipedia page "UK" is accessed, it will redirect the user to the "United Kingdom" page. This behavior of Wikipedia is possible thanks to the *redirect* metadata created by Wikipedia users when writing the articles. We extract the *redirect* metadata from the Wikipedia dump and build the *redirect dictionary* where each entry is in the form of $A \rightarrow B$ (A is redirected to B). The key A of each entry is converted to lowercase. We transform the plain text answer from the QA model to the corresponding Wikipedia entity by matching the answer with the key from *redirect dictionary* in the three following ways: (1) The answer is converted to lowercase (For example: "Isaac Newton" becomes "isaac newton"); (2) The answer is converted to lowercase and remove any punctuation (For example, "Isaac Newton," becomes "isaac newton"); (3) Only the capital words in the answer are used and converted to lowercase (For example: "by Isaac Newton" becomes "isaac newton"). In addition, (1) is applied before (2) because there are Wikipedia entities that contain punctuation, such as internet top-level domain ".ca" and ".us". If there is no matching key in the *redirect dictionary*, we use a simple BM25 approach to retrieve n closest Wikipedia entities from the list of all entities; In n entities, the entity with the lowest *Levenshtein distance* to the answer is selected. For questions asking about specific dates or numbers, we employ *regular expression* to extract numbers and different date formats from the answers and rearrange them following the format from the QA task in Zalo

Dataset	Train	Dev	Test	All
UIT-ViQuAD	18,579	2,285	2,210	23,074
VIMQA	8,041	1,003	1,003	10,047

Table 5.1: Overall of Vietnamese datasets for single-hop QA (UIT-ViQuAD) and multi-hop QA (VIMQA)

AI Challenge.

Model ensemble refers to techniques combining multiple models to produce better performance and plays an important role in an efficient end-to-end QA system. As our system consists of two separate steps, Retriever and Reader, we propose an ensemble method that combines multiple retrievers and readers to enhance the performance of the end-to-end system. Figure 5.6 shows our ensemble approach. For a set of m retrievers $\{Retr_1, Retr_2, ..., Retr_m\}$ and a set of n readers $\{Read_1, Read_2, ..., Read_n\}$, the question is first passed to m retrievers to retrieve m set of passages. The union of these passages is then used to create the final retrieval result containing k passages. The retrieval result is then passed to n readers and the Wikipedia-entity resolution module to obtain $k * n$ answer entities along with their confidence score. The same entities are grouped together, and their scores are summed. Finally, the entity with the highest summed score is chosen to be the final answer entity.

5.2 Experimental Result

5.2.1 Experimental Setup

Benchmarks and Knowledge Base

The proposed methods are evaluated on two large-scale human-generated Vietnamese QA Benchmarks, UIT-ViQuAD [41] for single-hop QA and VIMQA [29] for multi-hop QA. The statistics of the two datasets are shown in Table 5.1.

Knowledge Base

Most of the experiments are conducted using the Vietnamese Wikipedia Corpus as a knowledge base for QA. After the pre-processing step, the Wikipedia Corpus contains approximately 1,200,000 articles and almost 4,000,000 passages. We also conduct experiments on UIT-ViQuAD knowledge base [43],

	Vietnamese Wikipedia	**UIT-ViQuAD**
#articles	1,273,420	174
#passages	3,885,030	5,109

Table 5.2: Comparison of Vietnamese Wikipedia and UIT-ViQuAD knowledge base

Method	**Development Set**				**Test Set**			
	Top-1	**Top-5**	**Top-10**	**Top-20**	**Top-1**	**Top-5**	**Top-10**	**Top-20**
Lucene-BM25 [33]	36.96	55.09	61.72	66.94	30.43	47.78	54.17	60.28
DPR [25]	11.47	24.07	29.93	36.54	9.32	19.95	25.79	32.94
ViWiQA-Single Retriever (Ours)	**56.80**	**71.64**	**74.58**	**76.29**	**50.41**	**63.95**	**67.44**	**69.75**

Table 5.3: Top-k retrieval accuracy on UIT-ViQuAD development and test sets using Vietnamese Wikipedia knowledge base, measured by the proportion of top k passages retrieved containing the answer.

a minimal subset of the Vietnamese Wikipedia knowledge base, to compare ViWiQA with the work of Nguyen et al. [43]. Table 5.2 compares the Vietnamese Wikipedia and UIT-ViQuAD knowledge base.

The performance of end-to-end systems is measured by Exact Match (EM) and F1 score. We use the official evaluation script from SQuAD [49]. The EM score is measured by the parts of the predictions that exactly match the labeled answer spans after normalization and is formulated as follows.

$$\mathbf{EM} = \frac{\#\text{ exactly correct answers}}{\#\text{ questions}}$$

The F1 score is measured by Precision and Recall, which are formulated as follows.

$$\mathbf{Precision} = \frac{\#\text{ correctly predicted tokens}}{\#\text{ predicted tokens}}$$

$$\mathbf{Recall} = \frac{\#\text{ correctly predicted tokens}}{\#\text{ tokens in gold answer-span}}$$

$$\mathbf{F1} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

The ViWiQA-ER is evaluated on the public test set of the Zalo AI Challenge 2022 (End-to-end Question-Answering task). The test set consists of approximately 600 questions whose answers are a Wikipedia entity, a specific date, or a number. The evaluation is measured by the EM score.

Competitive Baselines

We adopt competitive methods in Open-domain QA to the Vietnamese QA datasets and use them as baselines for evaluation. The baselines include a sparse retriever Lucene-BM25 and a dense retriever DPR. Lucene-BM25 has been proven to be a competitive baseline to evaluate Open-domain QA systems. Especially in datasets like SQuAD, UIT-ViQuAD, or VIMQA, the question and the passages have a high number of overlapping tokens, giving BM25 a benefit. DPR is a retriever method that uses dense representations and establishes competitive results on various English Open-domain QA Dataset like NaturalQuestion [27], Trivia [24], WebQuestions [4], and TREC [2]. Moreover, the results of DrQA [6], BERTSerini [67], and XLMQA [43] on UIT-ViQuAD are taken from the work of Nguyen et al. [43] to compare with ViWiQA on the UIT-ViQuAD knowledge base.

Implementation Details

We employed Pyserini [33] to index the Vietnamese Wikipedia Corpus and build the Lucene-BM25 retriever. We follow the implementation of [64] to implement our Cross-Encoder and Reader models as described in section 5.1.3. XLM-RoBERTa$_{\text{Base}}$ and XLM-RoBERTa$_{\text{Large}}$ are used for the Cross-Encoder and the Reader, respectively. For ViWiQA-Single, we let $m = 100$ be the number of passages retrieved by Lucene-BM25, $n = 30$ be the number of top answer span predictions to keep.

To adapt DPR to Vietnamese, we employ the DPR implementation of Karpukhin et al. [25] and follow the DPR paper to train its dual-encoders using the UIT-ViQuAD dataset and the Vietnamese Wikipedia knowledge base. Because DPR uses the in-batch negatives technique and the training batch sizes greatly impact the DPR performance, we experiment with different batch sizes and select the best results as the baseline. As the current DPR architecture only accept a variant of BERT, we employ the pre-trained model Multilingual-BERT in our DPR adaptation to Vietnamese.

5.2.2 Results

Single-hop Question-Answering

We conducted experiments to evaluate the proposed single-hop retriever and end-to-end QA system *ViWiQA-Single*. Table 5.3 shows a comparison of different retriever methods on UIT-ViQuAD development and test sets using Vietnamese Wikipedia knowledge base, measured by top-k accuracy ($k \in 1, 5, 10, 20$). The result indicates that ViWiQA-Single retriever

System	**Development Set**		**Test Set**	
	EM	**F1**	**EM**	**F1**
Lucene-BM25	27.61	39.66	21.33	34.59
DPR (Batch 96)	1.05	0.00	1.13	0.00
ViWiQA (Ours)	**42.93**	**57.22**	**36.37**	**52.95**

Table 5.4: End-to-end single-hop QA performance on UIT-ViQuAD using the Wikipedia knowledge base. Lucene-BM25 and DPR are adapted to Vietnamese Wikipedia from the cited papers. The Reader model of these systems is XLM-RoBERTa$_{\text{Large}}$.

System	**EM**	**F1**
DrQA [6]	17.87	37.37
BERTserini [67]	36.52	55.55
XLMRQA [43]	47.96	61.83
ViWiQA-Single (Ours)	**53.98**	**69.59**

Table 5.5: End-to-end QA performance on UIT-ViQuAD test set using UIT-ViQuAD knowledge base.

performs consistently and significantly better than Lucene-BM25 and DPR. The performance gap is especially large when k is small. The top-1 accuracy of the proposed method is higher than Lucene-BM25 by about 20%. The experiment also suggests that DPR suffers greatly in multilingual and low data situations, complementing the observation of Gao et al. [19] and Ren et al. [53] about the limitation of dense representation via dual-encoders.

Table 5.4 shows the end-to-end evaluation on UIT-ViQuAD. The result suggests that better retrieval accuracy can enhance the end-to-end QA results and that ViWiQA-Single significantly outperforms Lucene-BM25 by about 15% and 18% absolute in EM and F1 scores, respectively.

We also conduct experiments over the UIT-ViQuAD knowledge base, a minimal subset of the Vietnamese Wikipedia knowledge base, to compare ViWiQA-Single with the work of Nguyen et al. [43]. Table 5.5 compares different systems over the UIT-ViQuAD knowledge base. The result of DrQA and BERTserini is derived from the research of Nguyen et al. [43]. The result indicates that ViWiQA-Single greatly outperforms other QA systems on the single-hop Vietnamese QA dataset.

Method	**Development Set**			**Test Set**		
	1C	**2C**	**CA**	**1C**	**2C**	**CA**
BM25	**58.72**	5.78	83.78	**55.23**	4.59	84.50
ViWiQA (Ours)						
ViWiQA-Single	42.87	2.69	84.64	41.38	3.09	85.49
ViWiQA-Multi	44.97	**9.27**	**86.20**	42.17	**8.18**	**86.63**

Table 5.6: Multi-hop retrieval accuracy over the Wikipedia knowledge base, evaluated on VIMQA dataset, measured in the percentage of retrieved passage pairs that have at least one correct passage (1C), has two correct passages (2C) or contains the answer (CA). Yes/No questions are excluded when measuring CA accuracy.

Multi-hop Question-Answering

This section evaluates the proposed multi-hop retriever and end-to-end multi-hop QA system *ViWiQA-Multi*. Table 5.6 shows the multi-hop retrieval accuracy over the Vietnamese Wikipedia knowledge base. The accuracy is measured using three metrics: the percentage of retrieved passage pairs with at least one passage whose title is correct (1C), the percentage of retrieved passage pairs with two passages whose titles are correct (2C), and the percentage of retrieved passage pairs that contain the answer span excluding Yes/No questions (CA). A title is considered correct if it is the same as the title of the gold passage in the dataset. Despite having lower accuracy in metric (1C) compared to Lucene-BM25, ViWiQA-Multi retriever has higher accuracy in metrics (2C) and (CA). The result indicates that the knowledge from the Wikipedia hyperlink graph successfully enhances the multi-hop retriever model in retrieving the correct passage pairs. In contrast, while Lucene-BM25 has a high percentage of retrieving one correct passage in the passage pair, it struggles to find both correct passages. It is essential for multi-hop questions that both passages in the retrieved pair are correct so that the Reader model can perform multi-hop reasoning and find the answer.

Table 5.7 shows the end-to-end evaluation on VIMQA. The result indicates that the proposed system significantly outperforms Lucene-BM25 with XLM-R Reader by about 5% to 6% in EM accuracy and 7% to 10% in F1 score. ViWiQA-Multi also performs better than ViWiQA-Single, indicating the improvement that the Wikipedia hyperlink graph has on the multi-hop retrieval process.

Method	Development Set		Test Set	
	EM	F1	EM	F1
Lucene-BM25	35.29	44.32	35.99	45.99
ViWiQA (Ours)				
ViWiQA-Single	38.78	50.10	39.08	51.42
ViWiQA-Multi	**41.77**	**53.97**	**40.98**	**53.33**

Table 5.7: End-to-end evaluation of multi-hop QA systems on VIMQA measured in Exact Match (EM) and F1 Score

Wikipedia-Entity-Level Question-Answering

We conduct experiments on the Zalo AI Challenge 2022 (End-to-end Question-Answering task) public test set to evaluate the proposed ViWiQA-ER system. Table 5.8 shows the performance of ViWiQA-ER on the public test set, measured in EM score. The first group uses Lucene-BM25 [33] as the retriever, and the second group uses ViWiQA-Single Retriever as the retriever. Both groups use XLM-RoBERTa-Large[10] as the reader. The result indicates the efficiency that the proposed Wikipedia-Entity resolution module and the model ensemble method have on the performance of ViWiQA-ER. Approximately 8% and 10% absolute gain in EM score can be observed when applying Wikipedia-Entity resolution and Number/Date processing, respectively. The proposed ensemble method successfully boosts the system with about 2% absolute gain in EM score. Our ViWiQA-ER system achieved the **2$^{\text{nd}}$ Place** in the official **final result of Zalo AI Challenge 2022** (End-to-end Question-Answering task).

Ablation Study

The first ablation study investigates the effectiveness of existing pre-trained models when applied to the Vietnamese QA reader. Table 5.9 compares the performance of different pre-trained Transformer reader models on the task of QA with given contexts. We compare pre-trained models on Vietnamese corpus PhoBERT [37] and multilingual corpus Multilingual-BERT [14], XLM-RoBERTa [10]. The *Base* and *Large* versions of each pre-trained model are used, except for Multilingual-BERT, which is only available in *Base* version. The result suggests that XLM-RoBERTa is the most effective model when adapted to Vietnamese, outperforming the Vietnamese pre-trained model PhoBERT.

Retriever	Reader	Wikipedia ER	Number/Date	Ensemble	EM
BM25	XLM-RoBERTa-Large				55.33
		✓			62.67
			✓		62.67
				✓	56.33
		✓	✓		70.00
		✓	✓	✓	71.00
ViWiQA-Single	XLM-RoBERTa-Large				61.33
		✓			68.83
			✓		71.00
				✓	63.00
		✓	✓		78.50
		✓	✓	✓	**80.50**

Table 5.8: Evaluation of ViWiQA-ER on Zalo AI Challenge 2022 (End-to-end QA task) public test set.

Reader	**EM**	**F1**
PhoBERT$_{\text{Large}}$	64.17	82.45
PhoBERT$_{\text{Base}}$	60.19	79.56
Multilingual-BERT$_{\text{Base}}$	61.19	81.02
XLM-RoBERTa$_{\text{Large}}$	**72.83**	**89.73**
XLM-RoBERTa$_{\text{Base}}$	66.94	85.19

Table 5.9: Comparing different Vietnamese and multilingual pre-trained Transformer models on UIT-ViQuAD test set (QA with given contexts task).

The second ablation study explores the in-batch negatives technique of DPR when adapted to Vietnamese. Figure 5.7 illustrates the influence of different training batch sizes on the Top-k retrieval accuracy of DPR. Owing to limited computational resources, the largest batch size in the experiment is 96. The result suggests that training batch size affects the performance of DPR to an extent. However, as shown in section 5.2.2, the result at batch size 96 is still significantly lower than Lucene-BM25, indicating the DPR limitation when adapted to low-resource language.

We perform an ablation study focusing on analyzing the method in the ViWiQA-Single retriever. Table 5.10 shows the performance of ViWiQA-Single retriever in three different settings. The first setting only uses the relevance scores from the Cross-Encoder model to rerank the passages. The second setting only uses the top answer span predictions from the Reader model to filter the passages. The third setting is the full method where both Cross-Encoder and Reader outputs are considered, as in Section 5.1.3.

Method	Development Set				Test Set			
	Top-1	Top-5	Top-10	Top-20	Top-1	Top-5	Top-10	Top-20
ViWiQA-Single Retriever (CE)	55.62	70.72	74.41	76.25	49.18	63.09	66.76	69.25
ViWiQA-Single Retriever (RE)	39.51	58.65	64.49	70.54	32.70	51.18	57.79	64.18
ViWiQA-Single Retriever (Full)	**56.80**	**71.64**	**74.58**	**76.29**	**50.41**	**63.95**	**67.44**	**69.75**

Table 5.10: Top-k retrieval accuracy on UIT-ViQuAD development and test sets using Vietnamese Wikipedia knowledge base, measured by the proportion of top k retrieval results containing the answer. ViWiQA-Single is evaluated in three settings: Only rerank using the Cross-Encoder (CE), only filter by the predictions of Reader (RE), and full method using both the CE scores and Reader predictions (Full). See text for more details.

System	Development Set		Test Set	
	EM	F1	EM	F1
ViWiQA (CE)	42.14	55.98	35.46	51.64
ViWiQA (RE)	29.28	42.15	22.74	37.29
ViWiQA (Full)	**42.93**	**57.22**	**36.37**	**52.95**

Table 5.11: End-to-end single-hop QA performance on UIT-ViQuAD using the Wikipedia knowledge base. ViWiQA is evaluated using only the Cross-Encoder (CE), only Reader predictions (RE), and full method (Full).

Table 5.11 shows the evaluation of the end-to-end ViWiQA-Single in the same three settings. The result suggests that reranking passages using the Cross-Encoder model gives more performance gains than filtering passages using the Reader model. However, filtering passages using the Reader model still gives an essential improvement in the full method.

We also analyze specific retrieval results from Cross-Encoder and ViWiQA-Multi Retriever to grasp the merits and demerits of the models. Table 5.12 and 5.13 shows two retrieval examples using Cross-Encoder and ViWiQA-Multi. In these examples, the models aim to retrieve the passage pair to support answering the multi-hop question. The retrieval results indicate that although Cross-Encoder can retrieve relevant passages to the question, these passages are not connected in a way that facilitates multi-hop reasoning. As a result, the passage-pair from Cross-Encoder cannot support answering the multi-hop question. On the other hand, between the first and the second passage in the passage-pair retrieved by ViWiQA-Multi Retriever, there is a connection through *bridge* entity like *"Manchester United F.C"* and *"Warner Bros"*. The *bridge* entities in multi-hop QA are usually entities that connect the contexts and facilitate the creation of multi-hop question [69, 29]. Therefore, ViWiQA-Multi Retriever produces high-quality passage-pairs that can answer multi-hop questions. These examples indicate the effectiveness of using Wikipedia hyperlink graphs to select candidate passages that are connected to the first passage for the second passage retrieval.

Question: Biệt danh của một trong những câu lạc bộ ở Anh mà Fabien Alain Barthez từng chơi và đoạt huy chương là gì? *(What is the nickname of one of the clubs in England where Fabien Alain Barthez played and won a medal?)*

Answer: Quỷ đỏ *(The Red Devils)*

Cross-encoder retrieval
Title 1: Fabien Barthez
Passage 1 (From Wikipedia): Fabien Alain Barthez (; sinh ngày 28 tháng 6 năm 1971) là một cựu cầu thủ bóng đá người Pháp. Ông đã từng đoạt một số huy chương khi chơi ở vị trí thủ môn cho Marseille, Manchester United và cùng với đội tuyển bóng đá quốc gia Pháp giành chức vô địch tại World Cup 1998, Euro 2000 và lọt vào trận chung kết World Cup 2006. Ông cùng với Peter Shilton là 2 thủ môn giữ kỷ lục giữ sạch lưới nhất trong giải vô địch bóng đá thế giới, trong 10 trận. Ở câu lạc bộ, ông *(Fabien Alain Barthez (; born 28 June 1971) is a French former footballer. He won several medals while playing as a goalkeeper for Marseille, Manchester United and won the 1998 World Cup, Euro 2000, and reached the 2006 World Cup final with the France national football team. He and Peter Shilton are the two goalkeepers who hold the record for keeping the most clean sheets in the World Cup, in 10 matches. At club level, he)*
Title 2: West Ham United F.C.

Passage 2 (From Wikipedia): West Ham United Football Club là một câu lạc bộ bóng đá chuyên nghiệp Anh đặt trụ sở vùng phía đông thành phố London, thủ đô nước Anh. West Ham United đã 3 lần đoạt Cúp FA, 1 lần đoạt Cúp C2 châu Âu và 1 lần đoạt cúp Intertoto. Sân nhà của câu lạc bộ là sân vận động Olympic với sức chứa khoảng 60.000 khán giả. Biệt danh của câu lạc bộ là "The Irons" hoặc "The Hammers". Các đối thủ truyền thống của West Ham United là các câu lạc bộ cùng thành phố *(West Ham United Football Club is an English professional football club based in the east of London, the capital of England. West Ham United has won the FA Cup three times, the European Cup once, and the Intertoto Cup once. The club's home ground is the Olympic Stadium with a capacity of about 60,000 spectators. The club's nickname is "The Irons" or "The Hammers". West Ham United's traditional rivals are clubs from the same city)*

ViWiQA-Multi retrieval (ours)
Title 1: Fabien Barthez
Passage 1 (From Wikipedia): Fabien Alain Barthez (; sinh ngày 28 tháng 6 năm 1971) là một cựu cầu thủ bóng đá người Pháp. Ông đã từng đoạt một số huy chương khi chơi ở vị trí thủ môn cho Marseille, Manchester United và cùng với đội tuyển bóng đá quốc gia Pháp giành chức vô địch tại World Cup 1998, Euro 2000 và lọt vào trận chung kết World Cup 2006. Ông cùng với Peter Shilton là 2 thủ môn giữ kỷ lục giữ sạch lưới nhất trong giải vô địch bóng đá thế giới, trong 10 trận. Ở câu lạc bộ, ông *(Fabien Alain Barthez (; born 28 June 1971) is a French former footballer. He won several medals while playing as a goalkeeper for Marseille, Manchester United and won the 1998 World Cup, Euro 2000, and reached the 2006 World Cup final with the France national football team. He and Peter Shilton are the two goalkeepers who hold the record for keeping the most clean sheets in the World Cup, in 10 matches. At club level, he)*
Title 2: Manchester United F.C.

Passage 2 (From Wikipedia): Anh, giải đấu hàng đầu trong hệ thống bóng đá Anh. Với biệt danh "Quỷ Đỏ", câu lạc bộ được thành lập dưới tên Newton Heath LYR Football Club vào năm 1878, đổi tên thành Manchester United vào năm 1902 và chuyển đến sân vận động hiện tại, Old Trafford, vào năm 1910. Manchester United là một trong những câu lạc bộ thành công nhất tại Anh, giữ kỷ lục 20 lần vô địch bóng đá Anh, đoạt 12 Cúp FA, 5 Cúp Liên đoàn và giữ kỷ lục 21 lần đoạt Siêu cúp Anh. Câu lạc bộ đã giành *(England, the top league in the English football system. Nicknamed ”the Red Devils”, the club was founded as Newton Heath LYR Football Club in 1878, renamed Manchester United in 1902 and moved to its current stadium, Old Trafford, in 1910. Manchester United is one of the most successful clubs in England, holding a record 20 times English football championship, won 12 FA Cups, 5 League Cups, and holds the record of 21 times won the English Super Cup. The club won)*

Table 5.12: Example 1: Passage-pair retrieval using Cross-Encoder and ViWiQA-Multi. English translation is provided in *italic*.

Question: Công ty phát hành album Death Magnetic có trụ sở ở đâu? *(Where is the company that publishes the album Death Magnetic based?)*

Answer: Burbank

Cross-encoder retrieval
Title 1: Death Magnetic
Passage 1 (From Wikipedia): Death Magnetic là album phòng thu thứ 9 của ban nhạc heavy metal đến từ Mỹ Metallica, phát hành ngày 12 tháng 9 năm 2008 bởi Warner Bros. Records. Đây là album đầu tiên của nhóm có sự góp mặt của tay Bass Robert Trujillo, và nhà sản xuất Rick Rubin. Đây cũng là album phòng thu đầu tiên của Metallica hợp tác với Warner Bros. Records. Album này sau khi phát hành đã leo lên vị trí số 1 tại bảng xếp hạng Billboard 200 của Mỹ với 490.000 bản được tiêu thụ ngay tuần đầu tiên. Với thành *(Death Magnetic is the ninth studio album by American heavy metal band Metallica, released on September 12, 2008 by Warner Bros. Records. This is the group's first album to feature bassist Robert Trujillo, and producer Rick Rubin. This is also Metallica's first studio album in collaboration with Warner Bros. Records. This album after its release climbed to No. 1 on the US Billboard 200 chart with 490,000 copies sold in the first week. With this achievement,)*
Title 2: Danh sách album quán quân năm 2008 (Mỹ) *(List of number-one albums of 2008 (USA))*

Passage 2 (From Wikipedia): 2009. Cô là nữ ca sĩ nhạc đồng quê duy nhất đạt được thành tích này trong lịch sử bảng xếp hạng Billboard 200. Hiện Swift được xếp hạng 5 trong danh sách các nữ nghệ sĩ hát đơn có được album quán quân lâu nhất, ngang hàng với Mariah Carey và Whitney Houston. Một vài album quán quân khác trong vài tuần bao gồm "Sleep Through the Static"bởi Jack Johnson và album thứ 9 "Death Magnetic"của Metallica; cả hai đều đứng đầu bảng trong 3 tuần liên tiếp. Trong năm 2008 có ba album nhạc phim (soundtrack) (*2009. She is the only female country singer to achieve this feat in the history of the Billboard 200 chart. Currently, Swift is ranked 5th on the list of female solo artists with the longest number-one album, equal to Mariah Carey and Whitney Houston. A few other number-one albums within a few weeks include "Sleep Through the Static" by Jack Johnson and the ninth album "Death Magnetic" by Metallica; both topped the table for 3 consecutive weeks. In 2008 there were three soundtrack albums (soundtrack)*)

ViWiQA-Multi retrieval (ours)
Title 1: Death Magnetic
Passage 1 (From Wikipedia): Death Magnetic là album phòng thu thứ 9 của ban nhạc heavy metal đến từ Mỹ Metallica, phát hành ngày 12 tháng 9 năm 2008 bởi Warner Bros. Records. Đây là album đầu tiên của nhóm có sự góp mặt của tay Bass Robert Trujillo, và nhà sản xuất Rick Rubin. Đây cũng là album phòng thu đầu tiên của Metallica hợp tác với Warner Bros. Records. Album này sau khi phát hành đã leo lên vị trí số 1 tại bảng xếp hạng Billboard 200 của Mỹ với 490.000 bản được tiêu thụ ngay tuần đầu tiên. Với thành (*Death Magnetic is the ninth studio album by American heavy metal band Metallica, released on September 12, 2008 by Warner Bros. Records. This is the group's first album to feature bassist Robert Trujillo, and producer Rick Rubin. This is also Metallica's first studio album in collaboration with Warner Bros. Records. This album after its release climbed to No. 1 on the US Billboard 200 chart with 490,000 copies sold in the first week. With this achievement,*)
Title 2: Warner Bros.

Passage 2 (From Wikipedia): Công ty Giải Trí Warner Brothers (hay Warner Bros., Warner Bros. Pictures) là một trong những hãng sản xuất phim và truyền hình lớn nhất thế giới. Đây là một chi nhánh từ Time Warner, trụ sở đặt tại Burbank, California và New York City. Warner Bros. có vài công ty con khác như Warner Bros. Studios, Warner Bros. Pictures, Warner Bros. Games, Warner Bros. Television, Warner Bros. Animation, Warner Home Video, DC Comics và New Line Cinema. Warner chiếm lĩnh một nửa thị trường The CW Television Network. Được thành lập năm 1918 bởi những người nhập cư từ *(Warner Brothers Entertainment Company (or Warner Bros., Warner Bros. Pictures) is one of the largest film and television production companies in the world. This is an affiliate from Time Warner, with headquarters in Burbank, California and New York City. Warner Bros. has several other subsidiaries such as Warner Bros. Studios, Warner Bros. Pictures, Warner Bros. Games, Warner Bros. Television, Warner Bros. Animation, Warner Home Video, DC Comics, and New Line Cinema. Warner dominates half of The CW Television Network's market. Founded in 1918 by immigrants from)*

Table 5.13: Example 2: Passage-pair retrieval using Cross-Encoder and ViWiQA-Multi. English translation is provided in *italic.*

5.3 Summary

This work proposes efficient single and multi-hop retriever methods for Open-domain QA in Vietnamese over the Wikipedia knowledge base. The single-hop retriever utilizes relevance scores produced by the Cross-Encoder model and answer predictions from the Reader model to enhance the retrieval. The multi-hop retriever enhances the quality of retrieved passage pairs by integrating the Wikipedia hyperlink graph in the retrieval process. Using the proposed retrievers, we develop end-to-end Open-domain QA systems that achieve new state-of-the-art results in standard Vietnamese single and multi-hop QA datasets. The efficacy of the proposed systems can be confirmed by comparing the experimental results with strong baselines in resource-rich languages.

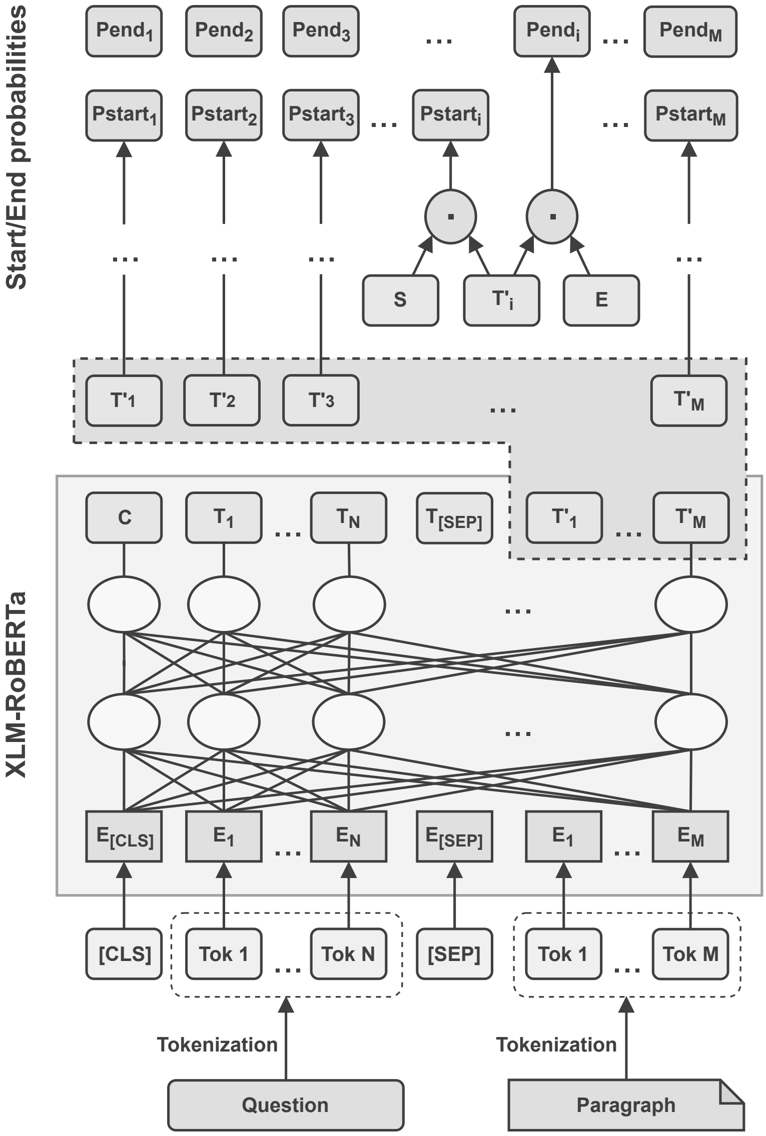


Figure 5.4: Detailed architecture of Reader model for predicting answer spans

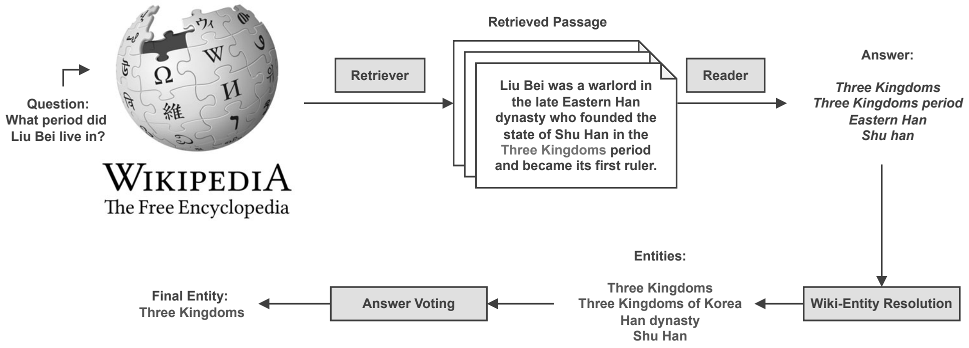


Figure 5.5: Overall end-to-end QA system with Wikipedia entity resolution.

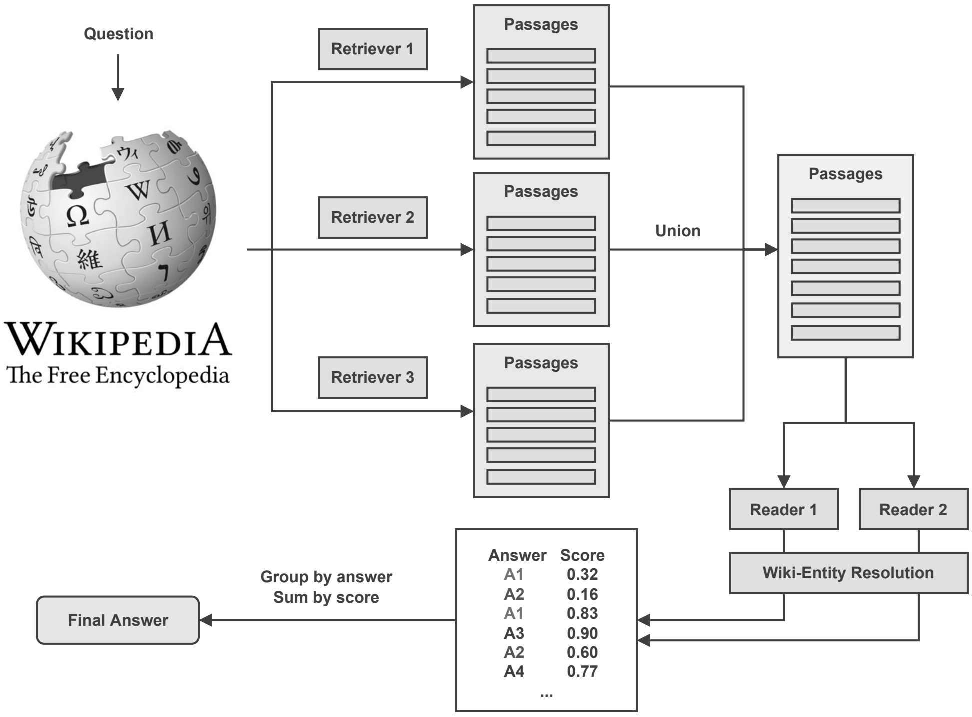


Figure 5.6: Ensemble approach for multiple retrievers and readers.

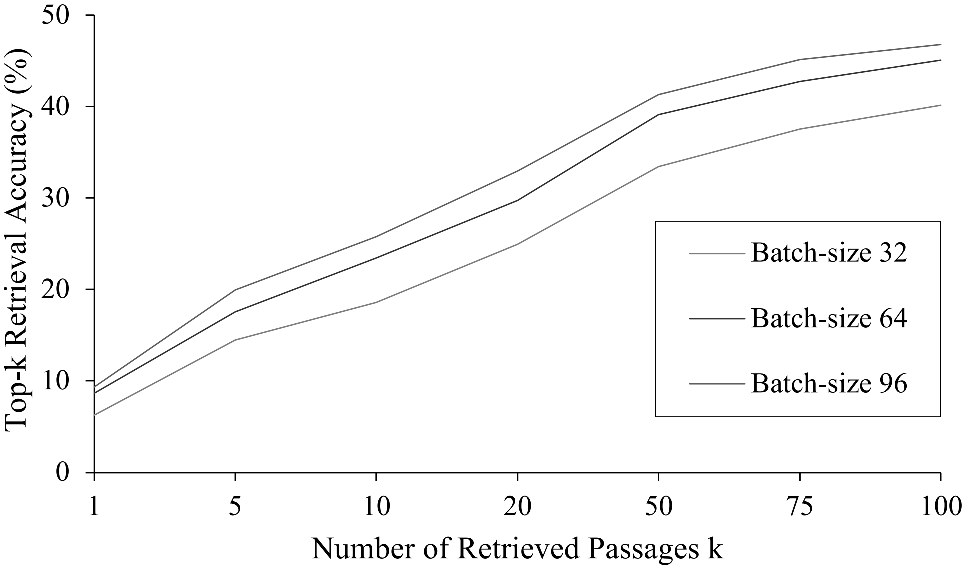


Figure 5.7: Retrieval accuracy of DPR on UIT-ViQuAD test set at different training batch sizes when adapted to Vietnamese Wikipedia, measured by the proportion of top k retrieval results containing the answer.

Chapter 6

Conclusions

6.1 Conclusions

This research develops an efficient Open-domain QA system over the Wikipedia knowledge base for single and multi-hop questions. The proposed system is robust when applied to low-resource languages. This research was initially conducted in the Vietnamese language, but the methodology can be generalized to other low-resource languages. To this end, the contributes VIMQA dataset and ViWiQA system.

- We present VIMQA, a multi-hop QA dataset for Vietnamese, and a method for gathering multi-hop examples that can be applied to other languages. Our evaluation of VIMQA demonstrates the efficiency of the pipeline. Results from experiments show VIMQA presents a challenge for various approaches in single and multi-hop QA. It highlights the usefulness of VIMQA as a resource for both Vietnamese and cross-lingual QA models, especially for reasoning and explaining text comprehension and coherence in Vietnamese multi-hop QA tasks.

- We propose ViWiQA, a Vietnamese QA system containing efficient single and multi-hop retrievers. The single-hop retriever utilizes relevance scores produced by the Cross-Encoder model and answer predictions from the Reader model to enhance the retrieval. The multi-hop retriever enhances the quality of retrieved passage pairs by integrating the Wikipedia hyperlink graph in the retrieval process. Using the proposed retrievers, we develop end-to-end Open-domain QA systems that achieve new state-of-the-art results in standard Vietnamese single and multi-hop QA datasets. The efficacy of the proposed systems can be confirmed by comparing the experimental results with strong baselines in resource-rich languages.

6.2 Published Works

6.2.1 Related to Main research

- Nguyen-Khang Le, **Dieu-Hien Nguyen**, Tung Le Thanh, and Minh Le Nguyen. “VIMQA: A Vietnamese Dataset for Advanced Reasoning and Explainable Multi-hop Question Answering”. *In: Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 6521–6529.* 2022.

- Nguyen-Khang Le, **Dieu-Hien Nguyen**, Thi-Thu-Trang Nguyen, Minh Phuong Nguyen, Tung Le, and Minh Le Nguyen. “A Novel Pipeline to Enhance Question-Answering Model by Identifying Relevant Information”. *In: SCIDOCA 2021 post-proceedings (Accepted)*

- **Dieu-Hien Nguyen**, Nguyen-Khang Le, and Minh Le Nguyen. “ViWiQA: Efficient End-to-end Vietnamese Wikipedia-based Open-domain Question-Answering Systems for Single-hop and Multi-hop Questions”. *(Submitted to Information Processing & Management Journal)*

6.2.2 Other publications

- **Dieu-Hien Nguyen**, Nguyen-Khang Le, and Minh Le Nguyen (2022). “Exploring Retriever-Reader Approaches in Question-Answering on Scientific Documents”. *In: Recent Challenges in Intelligent Information and Database Systems. ACIIDS 2022. Communications in Computer and Information Science, vol 1716.*

- Chau Nguyen, Nguyen-Khang Le, **Dieu-Hien Nguyen**, Minh Phuong Nguyen, and Minh Le Nguyen (2022). “A Legal Information Retrieval System for Statute Law”. *In: Recent Challenges in Intelligent Information and Database Systems. ACIIDS 2022. Communications in Computer and Information Science, vol 1716.*

- Chau Nguyen, Minh-Quan Bui, Dinh-Truong Do, Nguyen-Khang Le, **Dieu-Hien Nguyen**, Thu-Trang Nguyen, Ha-Thanh Nguyen, Vu Tran, Le-Minh Nguyen, Ngoc-Cam Le, Thi-Thuy Le, Minh-Phuong Nguyen,Tran-Binh Dang, Truong-Son Nguyen, Viet-Anh Phan, Thi-Hai-Yen Vuong, Minh-Tien Nguyen, Tung Le, and Tien-Huy Nguyen, “ALQAC 2022: A Summary of the Competition”. *In: 2022 14th International Conference on Knowledge and Systems Engineering (KSE).* 2022, pp. 1-5.

- Quan Minh Bui, Chau Nguyen, Dinh-Truong Do, Nguyen-Khang Le, **Dieu-Hien Nguyen**, Thi-Thu-Trang Nguyen, Minh-Phuong Nguyen, and Minh Le Nguyen. “JNLP team: Deep Learning Approaches for Tackling Long and Ambiguous Legal Documents in COLIEE 2022”. *In: JURISIN 2022 post-proceedings (LNAI) (Accepted)*

- Nguyen-Khang Le, **Dieu-Hien Nguyen**, and Minh Le Nguyen. “An Effective Description Augmentation Approach for Visual Question Answering in Mathematics Abstract Diagram”. *(Submitted to The 32nd International Joint Conference on Artificial Intelligence (IJCAI) 2023)*

Bibliography

[1] Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *International Conference on Learning Representations*, 2020.

[2] Petr Baudiš and Jan Šedivý. Modeling of the question answering task in the yodaqa system. In *Proceedings of the 6th International Conference on Experimental IR Meets Multilinguality, Multimodality, and Interaction - Volume 9283*, CLEF'15, page 222–228, Berlin, Heidelberg, 2015. Springer-Verlag.

[3] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150, 2020.

[4] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

[5] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. In *ACL*, 2017.

[6] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[7] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[8] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3576–3588, Online, June 2021. Association for Computational Linguistics.

[9] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*, 2020.

[10] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics.

[11] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzman, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. pages 8440–8451, 01 2020.

[12] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics.

[13] Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. A dataset of information-seeking questions and answers anchored in research papers. In *NAACL*, 2021.

[14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina 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)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[17] Nell K. Duke and P. David Pearson. Effective practices for developing reading comprehension. *Journal of Education*, (1-2):107–122, 2009.

[18] D. A. Ferrucci. Introduction to "this is watson". *IBM Journal of Research and Development*, 56(3):235–249, may 2012.

[19] Luyu Gao and Jamie Callan. Condenser: a pre-training architecture for dense retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 981–993, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[20] Natural Language Computing Group. R-net: Machine reading comprehension with self-matching networks. May 2017.

[21] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR, 13–18 Jul 2020.

[22] Sanda Harabagiu, Dan Moldovan, Christine Clark, Mitchell Bowden, John Williams, and Jeremy Bensley. Answer mining by combining extraction techniques with abductive reasoning. pages 375–382, 01 2003.

[23] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, number 1398, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE.

[24] Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[25] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online, November 2020. Association for Computational Linguistics.

[26] Hooshang Khoshsima and Forouzan Tiyar. The effect of summarizing strategy on reading comprehension of iranian intermediate efl learners. *International Journal of Language and Linguistics*, 2:134–139, 01 2014.

[27] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019.

[28] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. 2020.

[29] Khang Le, Hien Nguyen, Tung Le Thanh, and Minh Nguyen. Vimqa: A vietnamese dataset for advanced reasoning and explainable multihop question answering. In *Proceedings of the Language Resources and Evaluation Conference*, pages 6521–6529, Marseille, France, June 2022. European Language Resources Association.

[30] Jinhyuk Lee, Seongjun Yun, Hyunjae Kim, Miyoung Ko, and Jaewoo Kang. Ranking paragraphs for improving answer recall in open-domain question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 565–569, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[31] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. pages 6086–6096, 01 2019.

[32] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020.

[33] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2356–2362, New York, NY, USA, 2021. Association for Computing Machinery.

[34] Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Wei Chen. Reader-guided passage reranking for open-domain question answering. In *ACL-IJCNLP 2021*, December 2020.

[35] Rada Mihalcea and Paul Tarau. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[36] Dan Moldovan, Marius Paşca, Sanda Harabagiu, and Mihai Surdeanu. Performance issues and error analysis in an open-domain question answering system. 21(2):133–154, apr 2003.

[37] Dat Quoc Nguyen and Anh Tuan Nguyen. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042, Online, November 2020. Association for Computational Linguistics.

[38] Kiet Nguyen, Nhat Nguyen, Phong Do, Anh Nguyen, and Ngan Nguyen. Vireader: A wikipedia-based vietnamese reading comprehension system using transfer learning. *Journal of Intelligent and Fuzzy Systems*, pages 1–19, 07 2021.

[39] Kiet Nguyen, Nhat Nguyen, Phong Do, Anh Nguyen, and Ngan Nguyen. Vireader: A wikipedia-based vietnamese reading comprehension system using transfer learning. *Journal of Intelligent and Fuzzy Systems*, 41:1–19, 07 2021.

[40] Kiet Nguyen, Vu Nguyen, Anh Nguyen, and Ngan Nguyen. A Vietnamese dataset for evaluating machine reading comprehension. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2595–2605, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[41] Kiet Nguyen, Vu Nguyen, Anh Nguyen, and Ngan Nguyen. A Vietnamese dataset for evaluating machine reading comprehension. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2595–2605, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[42] Kiet Nguyen, Vu Nguyen, Anh Nguyen, and Ngan Nguyen. A Vietnamese dataset for evaluating machine reading comprehension. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2595–2605, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[43] Kiet Van Nguyen, Phong Nguyen-Thuan Do, Nhat Duy Nguyen, Tin Van Huynh, Anh Gia-Tuan Nguyen, and Ngan Luu-Thuy Nguyen. XLMRQA: open-domain question answering on vietnamese wikipedia-based textual knowledge source. *CoRR*, abs/2204.07002, 2022.

[44] Rodrigo Frassetto Nogueira and Kyunghyun Cho. Passage re-ranking with BERT. *CoRR*, abs/1901.04085, 2019.

[45] Thanapon Noraset, Lalita Lowphansirikul, and Suppawong Tuarob. Wabiqa: A wikipedia-based thai question-answering system. *Information Processing I& Management*, page 102431, 2021.

[46] Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. Understanding the behaviors of BERT in ranking. *CoRR*, abs/1904.07531, 2019.

[47] Tran Minh Quan, David Grant Colburn Hildebrand, and Won-Ki Jeong. Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics. *Frontiers in Computer Science*, May 2021.

[48] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[49] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[50] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[51] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[52] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. pages 3973–3983, 01 2019.

[53] Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. RocketQAv2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2825–2835, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[54] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, apr 2009.

[55] Pum-Mo Ryu, Myung-Gil Jang, and Hyunki Kim. Open domain question answering using wikipedia-based knowledge model. *Inf. Process. Manag.*, pages 683–692, 2014.

[56] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. 11 2016.

[57] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30, pages 6000–6010. Curran Associates, Inc., 2017.

[58] Ellen M. Voorhees and Dawn M. Tice. The TREC-8 question answering track. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece, May 2000. European Language Resources Association (ELRA).

[59] Shuohang Wang and Jing Jiang. Learning natural language inference with lstm. 12 2015.

[60] Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. R3: Reinforced ranker-reader for open-domain question answering. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[61] Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati, and Bing Xiang. Multi-passage BERT: A globally normalized BERT model for open-domain question answering. 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)*, pages 5878–5882, Hong Kong, China, November 2019. Association for Computational Linguistics.

[62] Dirk Weissenborn, Georg Wiese, and Laura Seiffe. Fastqa: A simple and efficient neural architecture for question answering. 2017.

[63] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf,

Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

[64] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

[65] Wenhan Xiong, Xiang Lorraine Li, Srinivasan Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oğuz. Answering complex open-domain questions with multi-hop dense retrieval. *International Conference on Learning Representations*, 2021.

[66] Ming Yan, Chenliang Li, Chen Wu, Bin Bi, Wei Wang, Jiangnan Xia, and Luo Si. IDST at TREC 2019 deep learning track: Deep cascade ranking with generation-based document expansion and pre-trained language modeling. In Ellen M. Voorhees and Angela Ellis, editors, *Proceedings of the Twenty-Eighth Text REtrieval Conference, TREC 2019, Gaithersburg, Maryland, USA, November 13-15, 2019*, volume 1250 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2019.

[67] Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. End-to-end open-domain question answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[68] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive

pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[69] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[70] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[71] Adams Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc Le. Qanet: Combining local convolution with global self-attention for reading comprehension, 04 2018.

[72] Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. Retrieving and reading: A comprehensive survey on open-domain question answering. *CoRR*, abs/2101.00774, 2021.

[73] Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China, August 2021. Chinese Information Processing Society of China.