Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 8, No. 2, April, 2024

Multilingual Question Answering for Malaysia History with Transformer-based Language Model

Qi Zhi Lim [1], Chin Poo Lee [1*], Kian Ming Lim [1], Jing Xiang Ng [1], Eric Khang Heng Ooi [1], Nicole Kai Ning Loh [1]

[1] *Faculty of Information Science and Technology, Multimedia University, Melaka, 75450 Malaysia.*

Abstract

In natural language processing (NLP), a Question Answering System (QAS) refers to a system or model that is designed to understand and respond to user queries in natural language. As we navigate through the recent advancements in QAS, it can be observed that there is a paradigm shift of the methods used from traditional machine learning and deep learning approaches towards transformer-based language models. While significant progress has been made, the utilization of these models for historical QAS and the development of QAS for Malay language remain largely unexplored. This research aims to bridge the gaps, focusing on developing a Multilingual QAS for history of Malaysia by utilizing a transformer-based language model. The system development process encompasses various stages, including data collection, knowledge representation, data loading and pre-processing, document indexing and storing, and the establishment of a querying pipeline with the retriever and reader. A dataset with a collection of 100 articles, including web blogs related to the history of Malaysia, has been constructed, serving as the knowledge base for the proposed QAS. A significant aspect of this research is the use of the translated dataset in English instead of the raw dataset in Malay. This decision was made to leverage the effectiveness of well-established retriever and reader models that were trained on English data. Moreover, an evaluation dataset comprising 100 question-answer pairs has been created to evaluate the performance of the models. A comparative analysis of six different transformer-based language models, namely DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa, has been conducted, where the effectiveness of the models was examined through a series of experiments to determine the best reader model for the proposed QAS. The experimental results reveal that the proposed QAS achieved the best performance when employing RoBERTa as the reader model. Finally, the proposed QAS was deployed on Discord and equipped with multilingual support through the incorporation of language detection and translation modules, enabling it to handle queries in both Malay and English.

Keywords:

Question Answering;
Historical Knowledge;
Natural Language Processing;
DeBERTaV3;
BERT;
ALBERT;
ELECTRA;
MiniLM;
RoBERTa.

Article History:

Received:	23	November	2023
Revised:	24	February	2024
Accepted:	08	March	2024
Published:	01	April	2024

1- Introduction

History is referred to as the study and interpretation of past events, human activities, and societies. It encompasses the documentation, analysis, and understanding of the chronological sequence of events, causal relationships, and the evolution of cultures, civilizations, and institutions over time. The study of history serves multiple purposes. It provides insights into the achievements, struggles, and aspirations of past societies and offers valuable lessons and perspectives for contemporary societies. History allows us to understand the factors that have influenced the development of social, political, economic, and cultural systems. It sheds light on the interconnections and interdependencies between different regions and civilizations, helping people to recognize the shared heritage and global influences that have shaped this

[*] **CONTACT**: cplee@mmu.edu.my

DOI: http://dx.doi.org/10.28991/ESJ-2024-08-02-019

interconnected world. Additionally, history plays a crucial role in shaping collective memory and identity. It helps individuals and communities understand their roots, heritage, and cultural identity. By examining the past, a deeper understanding of origins can be acquired, the diversity of human experiences can be appreciated, and a sense of empathy and tolerance for different perspectives can be fostered. Unfortunately, from the surveys shown, students have low achievement in the history subject [1]. When students want to understand more about historical knowledge, various sources of data will be shown to them. It might cause confusion for the students, as some information might be wrong or appropriate [1]. Therefore, there is a need to create a question-answering system tailored to historical knowledge.

Question Answering System (QAS) is an application or technology that targets providing answers to user queries or questions in a human-like manner. It leverages the power of natural language processing (NLP) and information retrieval techniques to understand the meaning and intent of user questions. Later, the relevant information will be retrieved from a given knowledge base. Generally, a QAS is developed by following a three-step process: question understanding, information retrieval, and answer generation. For question understanding, the system will analyze the query to comprehend its meaning, intent, and context. This involves parsing the question, identifying keywords, extracting relevant information, and determining the type of answer expected. For information retrieval, the system will search for the relevant information within a knowledge base or a collection of documents. Various techniques, such as keyword matching, semantic search, and advanced information retrieval models, are employed to locate and rank relevant passages or documents. For answer generation, the system will process and synthesize the most appropriate answer to the user's question based on the retrieved information. This could involve extracting specific sentences or passages, summarizing relevant information, or generating a concise response using natural language generation techniques.

Based on the review of existing literature, it is evident that the methodology of QAS has evolved from traditional machine learning and deep learning approaches to predominantly utilizing transformer-based language models. This paradigm shift underscores the increasing significance and effectiveness of transformer architectures in enhancing the capabilities and performance of QAS. Despite the notable progress made in the field of QAS, there is still a noticeable research gap in which the ability of the models to address historical questions remains largely unexplored. Therefore, this research aims to bridge the gap by exploring the use of transformer-based language models for historical QAS. From another perspective, it can be found that only a limited number of existing works have explored the application of QAS in the Malay language. Hence, this research also emphasizes the development of a multilingual QAS with support for both Malay and English. More specifically, this study aims to develop a multilingual QAS specifically tailored to historical knowledge, with a particular focus narrowed down to the history of Malaysia.

The system development process involves multiple stages, including data collection, knowledge representation, data loading and pre-processing, document indexing and storing, and the establishment of the querying pipeline with the retriever and reader. In this study, a dataset comprising 100 articles, including web blogs related to the history of Malaysia, has been constructed, serving as the knowledge base for the proposed QAS. It is worth noting that the original dataset in Malay was translated into English to leverage the effectiveness of well-established retriever and reader models trained on English data. Additionally, an evaluation dataset consisting of 100 question-answer pairs has been created for the evaluation of the model's performance. This study explored six different transformer-based language models, namely DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa. A series of experiments have been conducted to determine the best reader model for the question-answering process. While the proposed QAS utilizes an English-based model and operates on an English dataset, it can provide answers in Malay for user presentation by incorporating language detection and translation modules. Finally, the scope of this research extends to the deployment of the proposed QAS on Discord.

The main contributions of this study are:

- Proposed a novel Multilingual QAS for Malaysian History by utilizing a transformer-based language model.
- Conducted a comparative analysis of six different transformer-based language models through extensive experiments using the self-collected evaluation dataset.
- Deployed the proposed QAS on Discord with multilingual support (Malay and English) through the incorporation of language detection and translation modules.

The rest of the paper is organized as follows: Section 2 presents a comprehensive literature review of QAS. Section 3 explains the methodology used to develop the proposed QAS in depth. Section 4 covers the experiment details and the experimental results. Section 5 describes the deployment of the proposed QAS. Finally, the conclusion and future work for this research are included in Section 6.

2- Literature Review

Earlier QAS, such as LUNAR [2], were merely a natural language front-end for structured database query systems. These systems utilized natural language processing (NLP) techniques to analyze questions posed by users and transform them into a canonical form. This canonical form was then used to construct a standard database query that could be understood by the underlying database system. Androutsopoulos et al. [3] introduced the Natural Language Interface to Databases (NLITB), which enables users to query databases using natural language instead of formal query language. It uses a domain-specific semantic grammar that maps natural language phrases to the corresponding database queries. The grammar used consists of production rules that define the syntactic and semantic structures of natural language expressions. Other than that, it also deploys conceptual graphs to represent the meaning of queries and database contents. Compared to LUNAR, NLTIB is more user-friendly as it lessens the use of formal query languages.

As time goes on, QAS has grown with the purpose of detecting intended question requirements in their natural form through the linguistic analysis of proposed questions. MASQUE [4] is a QAS that focuses on the linguistic analysis of the question asked. It represents queries in a logical form, typically using formal logic or logical representation language, to capture the intended meaning and requirements of the question. Once the question is represented in logical form, MASQUE then converts it into a database query for information retrieval. Zheng [5] presented an open-domain QAS, AnswerBus, based on sentence-level web information retrieval. In this research work, the system accepts queries from users using six languages, which are English, German, French, Spanish, Italian, and Portuguese, and provides answers to users using English. To answer the users' questions, five search engines and directories have been utilized to retrieve the relevant Web pages and extract sentences that are determined to contain answers.

While navigating through the latest advancements in QAS, it is evident that the development of QAS has been revolutionized by transformer-based language models. With the introduction of transformer architecture with attention mechanisms [6], researchers have explored novel approaches to QAS based on transformers. Ou et al. [7] proposed an automatic multimedia-based question-answer pair generation in the computer-assisted healthy education system using Mandarin. The system proposed is divided into three parts: the text generation module, the answer extraction module, and the BART-based question generation module. In this research, manually labeled question and answer pairs have been improved for the subsequent use of retrieval based QAS. From another perspective, Zhang [8] studied the application of similarity algorithms in designing an intelligent English QAS. In this study, the WordNet semantic dictionary is employed to assess the semantic information of a sentence to identify the longest word matched, then determine answers for the questions using the WordNet sentence similarity algorithm.

Das and Nirmala [9] enhanced healthcare QAS by adopting the BioBERT framework to identify suitable answers for the questions. The proposed healthcare QAS can be utilized for question generation and task-specific data that are related to the healthcare domain. Likewise, Gupta [10] studied the application of QAS in the biomedical domain and conducted a comparative analysis of various pretrained language models. Maximum Inner Product Search (MIPS) was utilized in the research to retrieve the top 10 passages for question answering. Pudasaini and Shakya [11] introduced a question-answering dataset for the biomedical domain and adopted transfer learning on the pretrained large language models. The results demonstrated the importance of domain-specific finetuning for the application of automated tasks in the biomedical field. Furthermore, Alzubi et al. [12] proposed COBERT, a dual algorithmic retriever-reader system for answering complex queries related to the Corona virus. The retriever employs the TF-IDF vectorizer while the reader utilizes BERT transformers, and the proposed DistilBERT model showcased outstanding performance across other pre-trained models in the specific question-answering task.

Archarya et al. [13] proposed a simple and smart QAS using Named Entity Recognition (NER) and BERT. In this work, NER is utilized to extract predefined keywords from the context, representing the most important part-of-speech of the data source. Following that, the BERT model is used to predict the answer to a question based on the predefined data. Similarly, Yin [14] researched QAS based on the BERT framework. BERT embeddings and a hierarchical attention model that comprises co-attention and self-attention mechanisms are employed to identify the consecutive paragraph range and generate answers to given questions. Yang et al. [15] proposed a knowledge graph question-answering (KGQA) model for bridge inspection. The proposed method enhanced the contextual representation through the combination of BERT and static domain dictionaries and resolved the problem of semantic matching by implementing the hierarchical cross-attention network. Moreover, Tian et al. [16] introduced an intelligent question-answering system for safety hazard knowledge based on deep semantic mining. In this study, BERT, Bidirectional Gated Recurrent Unit (BiGRU), and self-attention mechanisms have been integrated for effective feature extraction, and a Siamese neural network is implemented for answer selection.

Liu & Huang [17] developed a Chinese QAS based on GPT. In this research work, the sentences are not divided into words, but the whole sentence is utilized as input. This paper has contributed by replacing the language model module so that the Chinese context can be completely utilized by the Transformer. The researchers suggested that the QAS can be further enhanced by adding clustering to divide the problem and answer into different categories, which can boost the precision of the system and reduce computation complexity. Noraset et al. [18] proposed a novel Thai QAS, WabiQA, which was implemented using a BM25F-based document retriever and a bi-directional LSTM document reader. WabiQA utilized articles from Wikipedia as a knowledge source to perform question-answering in Thai language. The findings of this research work have proved that reading a small but relevant piece of text will benefit the overall question-answering performance.

Although numerous research studies have been conducted on QAS, there are only a limited number of works that explore the application of QAS in Malay language. Ainon et al. [19] proposed a Malay QAS named SIGMA, which comprises three main components: Parser, Analyzer, and Response Generator. The Parser transforms Bahasa Malaysia queries into semantic representations, while the Response Generator searches a domain-specific database for appropriate responses. These components utilize semantic grammar, transition trees, and ellipsis handling to process input requests efficiently and generate meaningful replies. Puteh et al. [20] explored Malay QAS on the Quran and emphasized the importance of question classification. The research presents a machine learning-based answer type classification model to identify the answer type, thereby assisting the QAS in retrieving the correct answers to the questions. Furthermore, Lim et al. [21] introduced an enhanced framework that combines translation models and convolutional neural networks (CNN) for effective question classification, improving the performance of Malay-English mixed-language QAS.

3- Methods

The development process of the proposed Question Answering System (QAS) in this study involves multiple key stages. The initial stage of the entire process is the data collection from the Internet and knowledge representation in separate text files. Next, the proposed QAS employs an indexing pipeline with a file converter and a preprocessor to load and pre-process the knowledge data. The documents are then indexed and stored into a Document Store, serving as the knowledge base for question-answering. Additionally, a querying pipeline is implemented to retrieve answers for user queries. Specifically, the querying pipeline consists of a retriever that retrieves relevant documents based on the query and a reader that extracts the most appropriate answers to the query.

As the main objective of this study is to build a QAS for historical knowledge, the impact of adopting different transformer-based language models as the reader model on question-answering performance has been investigated. Specifically, a total of six models are selected to be examined: DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa. It is worth noting that the implementation of the proposed system is done by utilizing the open-source Haystack framework [22]. Figure 1 depicts the workflow of the proposed QAS, highlighting the research methodology of this study. In the following subsections, each stage in developing the proposed QAS will be explained in detail to facilitate a comprehensive understanding.

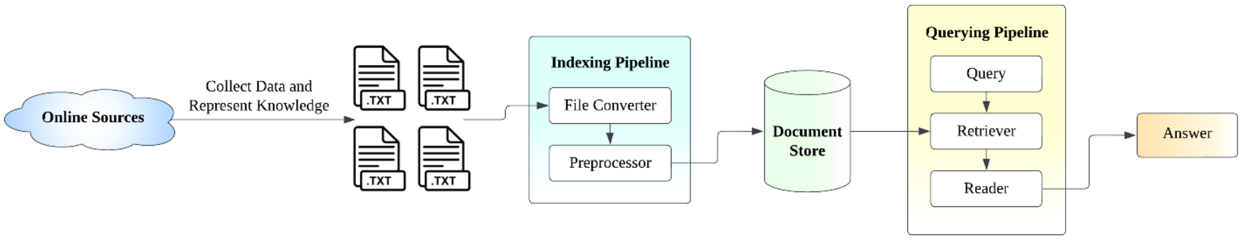


Figure 1. Question Answering System (QAS) Workflow

3-1- Data Collection and Knowledge Representation

In order to develop a QAS specifically tailored to historical knowledge, it is crucial to construct a dataset with diverse and representative historical content. This study primarily focuses on the history of Malaysia, aiming to serve as a proof of concept to demonstrate the effectiveness of the transformer-based language models in answering historical questions. As a result, a dataset comprising 100 articles, including web blogs that are written in Malay and related to the history of Malaysia, was collected from the Internet. During this phase, elements such as headers, footers, and references within the articles are removed to eliminate noise and streamline the content. The removal process was conducted manually by human annotators, ensuring the utmost accuracy in retaining only the relevant and informative content for further analysis and utilization. The contents of all the collected documents are then stored in separate text files, where each text file represents a single document.

Upon completing this stage, a collection of text files with contents related to Malaysia's history in Malay is produced. However, since the transformer-based language models selected for this study are extensively trained using English data, they specifically learn the inner representation of the English language that can be utilized in various downstream tasks. Hence, all the Malay text files are translated into English to create an English-version dataset to be used in the subsequent stages. This decision was made in order to align the language of the dataset with the language of the selected models, ensuring that the results accurately reflect the question-answering performance without being affected by the language constraints. In this study, the translator module utilized is the small variant of the BigBird transformer model [23], which is available through the Malaya library.

3-2- Indexing Pipeline

The indexing pipeline of the proposed QAS consists of two main components, which are a text converter and a preprocessor. With the knowledge representation in text files, the text converter in the indexing pipeline functions to load and convert the text files into Haystack document objects. Subsequently, the preprocessor in the indexing pipeline will pre-process the document objects in the system. The preprocessor follows several steps, starting with normalizing three or more consecutive empty lines to just two empty lines. Then, it removes leading and trailing whitespace from each line in the text. Additionally, it splits the documents into smaller segments, each containing a maximum of 200 words. This segmentation is crucial as it enhances the reader's scanning and extraction speed when processing the retrieved text and identifying the top answer candidates. In this work, the preprocessor is configured to split the documents in such a way that two adjacent documents overlap by 20 words, ensuring that the document boundaries do not fall in the middle of sentences.

By completing the pre-processing steps, a database, which is known as DocumentStore in Haystack, is initialized to index and store all the pre-processed document objects. In this particular work, the InMemoryDocumentStore is utilized, which is a simple document store that requires no external setup.

3-3- Querying Pipeline

Upon the completion of the indexing pipeline, the next stage in the proposed QAS is the querying pipeline. The querying pipeline is responsible for understanding and interpreting input queries, then retrieving the most relevant information from the indexed document store. The querying pipeline consists of two main components, which are the retriever and the reader. By utilizing the modules supported by the Haystack framework, the retriever and reader selected for the proposed QAS are the BM25 Retriever and FARM Reader.

Best Match 25 (BM25) Retriever is an information retrieval model that is implanted based on the BM25 ranking algorithm. The BM25 is a variation of Term Frequency-Inverse Document Frequency (TF-IDF), which outperforms its predecessor in two key aspects. The BM25 algorithm considers the term frequency, document length, and average document length in the collection to compute the relevance score of a document for a given query. On top of that, the ranking process of the BM25 is fine-tuned by incorporating factors such as term saturation and document length normalization. The BM25 Retriever in Haystack utilizes the BM25 scoring formula to rank the documents in the corpus based on their relevance to the query. The top-ranked documents are then passed to the reader of the querying pipeline for further processing.

Framework for Adapting Representation Models (FARM) Reader is a reader module supported by the Haystack framework, which is also developed by Deepset AI. The FARM Reader is specifically designed to extract answers for a query from a given context passage following a two-step process. First, a query-encoding approach will be used to embed both the query and the context passage into vector representations, which can be processed by machine learning models. Next, a prediction head will be applied on top of the encoded representations to generate answers for the specific query. The prediction head functions to predict the start and end positions of the answer span within the context passage. By identifying the boundaries of the answer span, the FARM Reader can extract the relevant text as the answer for a given query. The FARM Reader leverages powerful pre-trained transformer-based language models that have been trained on large-scale datasets and have a strong understanding of language semantics, enabling it to provide accurate and contextually relevant answers.

By utilizing the BM25 Retriever and FARM Reader, the complete querying pipeline for the proposed QAS is formed. When an input query is received during runtime, it will be passed to the BM25 Retriever to retrieve the top-ranked documents from the document store based on the relevance scores. By performing this, the scope of the search has been narrowed down, and the retrieved documents will be forwarded to the FARM Reader. Then, the reader will comprehend and analyze the content of the documents by applying the language models. The reader will search for relevant passages and assign confidence scores to each potential answer for the given query. Finally, the reader will select the answer with the highest confidence score and return it as the final result for the user presentation. With this, the entire workflow of the proposed QAS is uncovered.

3-3-1- Transformer-based Language Models for FARM Reader

In this study, a comparative analysis of six widely used transformer-based language models has been conducted to determine the best model to be employed as the reader for the proposed QAS. The models under consideration are DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa. These models were selected for their advanced transformer architecture, which enables them to comprehend complex language contexts and capture intricate patterns. Additionally, their pre-training on extensive English data ensures effective comprehension and precise responses in various NLP tasks, including question-answering.

It is worth noting that all the selected models are off-the-shelf models that have been fine-tuned for question-answering and are publicly available on Hugging Face. They were fine-tuned on a popular reading comprehension dataset, Stanford Question Answering Dataset 2.0 (SQuAD2.0) [24]. SQuAD2.0 contains a total of 100,000 question-answer pairs and over 50,000 unanswerable questions. In order to facilitate a better understanding, the selected models will be introduced and explained accordingly in the following paragraphs.

DeBERTaV3 [25] is an enhanced version of the Decoding-enhanced BERT with Disentangled Attention (DeBERTa) model [26], which improves the BERT and RoBERTa models by using the disentangled attention mechanism and an enhanced mask decoder. It introduces a new pre-training task known as Replaced Token Detection (RTD) to replace the conventional Mask Language Modeling (MLM) task. The research revealed that the sharing of embeddings in the ELECTRA model may cause conflicting directions in the token embeddings between the discriminator and the generator, leading to undesirable dynamics. To address the issue, a novel Gradient-Disentangled Embedding Sharing method is proposed in DeBERTaV3 to effectively avoid the conflicting dynamics, thereby improving the training efficiency and quality of the pre-trained model.

Bidirectional Encoder Representations from Transformers (BERT) [27] is a transformer model pre-trained on a large corpus of English data in a self-supervised manner. It incorporates two key objectives during the pre-training process, which are Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM involves randomly masking words in a sentence and training a BERT model to predict the masked words, enabling it to learn bidirectional representations. On the other hand, NSP focuses on predicting if two masked sentences are sequentially connected or not, enhancing the understanding of the BERT model in sentence relationships. By leveraging these objectives, the BERT model acquires a comprehensive understanding of the English language, allowing it to extract features and train classifiers for a wide range of downstream tasks.

A Lite BERT (ALBERT) [28] is an innovative language model that has made significant contributions to NLP. The main contributions of ALBERT include three key components, which are factorized embedding parameterization, cross-layer parameter sharing, and inter-sentence coherence loss. Factorized embedding parameterization reduces the memory requirements of the model by breaking down the large vocabulary embeddings into smaller matrices. Cross-layer parameter sharing enables the model to share parameters across layers, significantly reducing the model size without compromising performance. Lastly, the inter-sentence coherence loss introduced during pre-training enhances the model's ability to understand relationships between sentences in a document. These contributions not only made ALBERT more memory-efficient and computationally scalable but also improved its language understanding capabilities, leading to a revolution in the development of efficient and effective language models.

Efficiently Learning an Encoder that Classifies Token Replacements Accurately (ELECTRA) [29] is an advanced technique used for pre-training in NLP. Unlike conventional methods that primarily use masked language modeling, ELECTRA employs a unique framework consisting of a generator and a discriminator. The generator is trained to replace input tokens with plausible alternatives, while the role of the discriminator is to differentiate between the original and replaced tokens. This approach motivates the generator to generate realistic and contextually fitting replacements, leading to effective and efficient pre-training. With this, ELECTRA has showcased remarkable performance across various downstream tasks in the field of NLP.

MiniLM [30] is a novel language model developed by Microsoft Research Asia to address the challenges posed by the large parameter sizes of pre-trained models such as BERT in real-world NLP applications. It introduces a simple and effective approach to compress the large transformer models, named deep self-attention distillation. It involves training a smaller model, referred to as the student, to closely emulate the crucial self-attention module of the larger model, known as the teacher. Additionally, MiniLM enhances the distillation process by introducing the scaled dot-product between values in the self-attention module as new deep self-attention knowledge, along with the attention distributions. With these advancements, MiniLM overcomes the challenges associated with size and computational requirements, enabling more practical and efficient deployment in real-world NLP applications.

Robustly optimized BERT approach (RoBERTa) [31] is an advanced transformer language model that builds upon the BERT architecture. It enhances the architecture of the BERT model by integrating extensive training data, extending the training time, and employing refined training methods. It eliminates the next-sentence prediction task, prioritizing masked language modeling and employing dynamic masking. These improvements enable RoBERTa to grasp a greater understanding of context and vastly enhance its language representation abilities, resulting in outstanding performance across diverse NLP tasks and establishing it as a cutting-edge model in the field.

4- Experiments

To develop a promising Question Answering System (QAS) for historical knowledge, a series of experiments have been conducted in this study to find out the optimal reader model for determining the best answers to user queries based on the retrieved context. Specifically, six experiments have been carried out to study the effectiveness of employing different transformer-based language models as the reader model for the proposed QAS, namely DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa. Through a comparative analysis of the models' performance, the aim is to determine the best model that yields the most promising results in answering historical questions. The dataset constructed for model evaluation is introduced in Section 4-1; the evaluation metrics employed to assess the performance of the models are covered in Section 4-2; and the experimental results are presented in Section 4-3.

4-1- Evaluation Dataset

To evaluate the performance of the selected models, a question-answering dataset consisting of 100 question-answer pairs is constructed. This evaluation dataset was created by analyzing the articles collected from online resources (described in Section 3.2) thoroughly to identify important pieces of information that could be used to formulate questions. As a result, a total of 100 question-answer pairs were generated, covering different aspects of the historical contents of the entire corpus, including people, places, dates, and facts. In this stage, human evaluators play an important role in verifying the question-answer pairs, ensuring that the answers to the questions are correct and can be directly extracted from the original source.

It is crucial to note that the evaluation dataset is generated from the translated corpus, which is in English instead of Malay. As mentioned in the previous section, the reason for constructing the evaluation dataset using the English corpus is because the models adopted in this study are primarily based on English. Meanwhile, the aim of this study is to verify the ability of the models to effectively extract the correct answers from the entire corpus. Therefore, by aligning the language of the evaluation dataset with the language of the models, it can be ensured that the evaluation results accurately reflect the performance of the models in the question-answering task.

4-2- Evaluation Metrics

In this study, four evaluation metrics have been applied to effectively evaluate the performance of the transformer-based language models, which are F1-score, Jaccard similarity, cosine similarity, and semantic textual similarity. The calculations of all the metrics are done for each sample, then averaged by the total number of samples in the evaluation dataset. Each of these metrics provides different perspectives on examining the models' performance in question-answering tasks. By considering these metrics, a comprehensive understanding of the model's effectiveness can be gained, providing guidance to select the best reader model for the proposed QAS.

F1-score is a metric that is used to measure the overall effectiveness of a model in terms of precision and recall. To calculate the F1-score, the ground truth label (gold answer) and the predicted answer will be taken as input. Before the calculation, the ground truth label and the predicted answer will be converted into a sequence of tokens through word splitting. Next, the common tokens are identified by the intersection between them, representing the correct answer extracted by the model. Then, the true positives (TP), false positives (FP), and false negatives (FN) are computed based on the token counts. After that, the precision and recall are calculated, indicating the ratio of correctly identified answers to the total number of answers identified and the ratio of correctly identified answers to the total number of correct answers in the ground truth label. Finally, the F1-score for the model is calculated as the harmonic mean of precision and recall, giving equal weight to both metrics. The calculation of the F1-score can be written as:

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{1}$$

Jaccard similarity is also known as the Jaccard index or Jaccard coefficient, where the similarity between two sets will be measured. To calculate the Jaccard similarity, the ground truth label and the predicted answer will first be converted into sets of individual tokens. This is done to focus on the unique elements and ignore the order of the words. The Jaccard similarity is then calculated as the size of the intersection of the ground truth label set and predicted answer set divided by the size of their union. Mathematically, it can be represented as:

$$Jaccard(A, B) = \frac{A \cap B}{A \cup B} \tag{2}$$

where A denotes the ground truth label set and B denotes the predicted answer set. The resulting Jaccard similarity value ranges from 0 to 1, where 0 indicates no overlap between the sets and 1 indicates a perfect match. It is worth noting that Jaccard similarity solely focuses on the shared elements between the ground truth and predicted answer sets and does not consider the exact word-by-word matching or the semantic understanding.

Cosine similarity is a common metric used in natural language processing (NLP) tasks to measure the similarity between two vectors representing textual information, which is also applied to question answering. To calculate the cosine similarity, the ground truth label and predicted answer are converted into vector form using the Bag-of-Words (BoW) model. Next, the dot product of the two vectors is computed. After that, the Euclidean magnitudes of both vectors are calculated, which is the square root of the sum of the squares of all its elements. Finally, the cosine similarity score is calculated by dividing the dot product of the two vectors by the product of the magnitudes of the two vectors. The formula for cosine similarity is as follows:

$$Cosine(X, Y) = \frac{X \cdot Y}{\|X\| \times \|Y\|} \tag{3}$$

where X denotes the ground truth label vector and Y denotes the predicted answer vector. The resulting cosine similarity score ranges from -1 to 1, where -1 indicates that the two vectors are diametrically opposed, 0 signifies that the vectors are orthogonal, and 1 means they are identical. In the context of question-answering, a higher cosine similarity score indicates greater similarity between the ground truth label and the predicted answer.

Semantic similarity is the measurement of the degree of similarity or relatedness between two or more texts based on their underlying meaning or semantics. To compute the semantic similarity between the ground truth label and the predicted answer, the sentence transformer is utilized. It is used to transform sentences into dense vector representations or embeddings in semantic space. For the evaluation in this study, a sentence transformer model named "all-MiniLM-L6-v2" is being used for the encoding task. To get the score for semantic similarity, the cosine similarity for the embeddings of the ground truth label and predicted answer is calculated. The resulting semantic similarity score ranges from 0 to 1, where 0 indicates completely dissimilar and 1 indicates completely similar. By quantifying the similarity between texts based on their underlying meaning, semantic similarity provides a more nuanced understanding of text relationships when evaluating the performance of a model in question-answering tasks.

4-3- Experimental Results

In this stage, extensive experiments were conducted using the evaluation dataset to assess the performance of different transformer-based language models. It is worth noting that the top-k parameter for the retriever is set to 3 to retrieve the three most relevant documents for the reader throughout all the experiments. On the other hand, the top-k parameter for the reader is set to 1, which means that it will only return the most appropriate answers to the questions. Table 1 summarizes the experimental results for all the models presented in this study, namely DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa.

Table 1. Summary of Experimental Results

Model	Average F1-score	Average Jaccard Similarity	Average Cosine Similarity	Average Semantic Similarity
DeBERTaV3	0.74	0.69	0.75	0.83
BERT	0.77	0.72	0.78	0.85
ALBERT	0.78	0.72	0.79	0.86
ELECTRA	0.78	0.72	0.79	0.86
MiniLM	0.80	0.74	0.81	0.88
RoBERTa	0.92	0.88	0.93	0.96

The experimental results presented in Table 1 clearly demonstrate the substantial impact of employing different reader models on the performance of the proposed QAS. From Table 1, it can be noticed that RoBERTa shows the most promising result by achieving an average F1-score of 0.92, an average Jaccard similarity of 0.88, an average cosine similarity of 0.93, and an average semantic similarity of 0.96. The outstanding performance of the RoBERTa model can be attributed to its deep contextual understanding, extensive pre-training, and effective handling of complex language structures, enabling it to better process and comprehend the textual data in the question-answering task.

In contrast, the performance of DeBERTaV3 as the reader model in the querying pipeline is the worst among all the models presented in the experiments conducted. The model only achieves 0.74, 0.69, 0.75, and 0.83 for the average F1-score, average Jaccard similarity, average cosine similarity, and average semantic similarity, respectively. The poor performance of the DeBERTaV3 model might be due to its limitations on contextual understanding as well as the intricate model architecture that does not align well with the question-answering task presented in this study. In the meantime, the BERT model performs slightly better than the DeBERTaV3 model, scoring at 0.77, 0.72, 0.78, and 0.85 for the evaluation metrics used in this study.

On the other hand, the experimental results show that ALBERT and ELECTRA have similar efficiency in identifying the best answer candidate from the retrieved documents. Both models have achieved an average F1-score of 0.78, an average Jaccard similarity of 0.72, an average cosine similarity of 0.79, and an average semantic similarity of 0.86. Moreover, while not as sophisticated as the RoBERTa model, MiniLM displayed decent efficiency in the question-answering task in this study, attaining 0.80, 0.72, 0.81, and 0.88 for the average F1-score, average Jaccard similarity, average cosine similarity, and average semantic similarity, respectively.

To sum up, the reader initialized with the RoBERTa model stands out as the best performer of all the other models presented in this study. To provide a more intuitive insight into the model's performance in answering historical questions, several samples from the evaluation dataset with the gold (correct) answer and the predicted answer are presented in Table 2. The samples are grouped into people, places, dates, and facts, showing that the RoBERTa model can accurately provide answers for different types of historical questions.

Table 2. Samples with Gold Answer and Predicted Answer (RoBERTa)

Types	Question	Gold Answer	Predicted Answer
People	Who made the decision for Singapore's separation from Malaysia?	Tunku Abdul Rahman Putra Al-Haj	Tunku Abdul Rahman Putra Al-Haj
	Who introduced the policy of applying Islamic values in the administration of Malaysia?	Mahathir Mohamad	Mahathir Mohamad
	Who signed the Federation of Malaya Agreement on behalf of King George VI?	Sir Edward Gent	Edward James Gent
	Who ruled Sarawak between 1841 and 1946?	The Brooke family	The Brooke family
	Who led the Naning residents during the war?	Dato 'Dol Said	Dato 'Dol Said
Places	Where did King Bagindo hail from?	Minangkabau	Minangkabau
	Where was Tan Cheng Lock born?	Heeren Road, Melaka	Heeren Road, Melaka
	Which school did Sambanthan attend in Kuala Kangsar, Perak?	Clifford School	Clifford School
	In which regions did the puppet appear in the Malay world, as mentioned by Dato' A. Aziz Deraman?	Kelantan and Java Island	Kelantan and Java Island
	What region is the Maya civilization located in?	Southern Mexico and northern Central America	Southern Mexico and northern Central America
Dates	When was Tun Dr. Mahathir bin Mohamad born?	July 10, 1925	July 10, 1925
	When was the Federation of Malaya Agreement signed?	January 21, 1948	January 21, 1948
	When did the Malaysian Constitution come into effect?	August 31, 1957	August 31, 1957
	When was ASEAN established?	August 8, 1967	August 8, 1967
	When was Rukun Negara formed?	31 August 1970	31 August 1970
Facts	What was the predecessor of the Federation of Malaya?	Malayan Union	Malayan Union
	What was the Great Revolution also known as?	The Revolution of 1688	The Revolution of 1688
	What are the three parts into which the power is divided in the Malaysian government?	Legislature, Justice, and Executive	Legislature, Justice, and Executive
	What are the three major races in Malaysia?	Malays, Chinese and Indians	Malays, Chinese and Indians
	What was the name of the agreement that combined North Borneo, Sarawak, and Singapore with the existing states in the Federation of Malaya?	Malaysia Agreement	Malaysia Agreement

By examining the samples provided in Table 2, it is more evident that the RoBERTa model has demonstrated its proficiency in effectively addressing historical questions in this study. Therefore, it is adopted as the reader model of the querying pipeline in developing the proposed QAS.

5- Deployment

Based on the experimental results, RoBERTa has been selected as the reader model for the proposed Question Answering System (QAS). The deployment of the proposed QAS is carried out on Discord, an online platform that provides a well-established channel for research and practical applications. Discord was chosen as the preferred platform to deploy the proposed QAS due to its user-friendly interface and large user base. By incorporating the language detection and translation modules, the proposed QAS can provide answers to questions in multiple languages, specifically English and Malay. Figure 2 shows a flowchart illustrating the process flow of the deployed QAS in Discord, providing a clearer insight into how the system works.

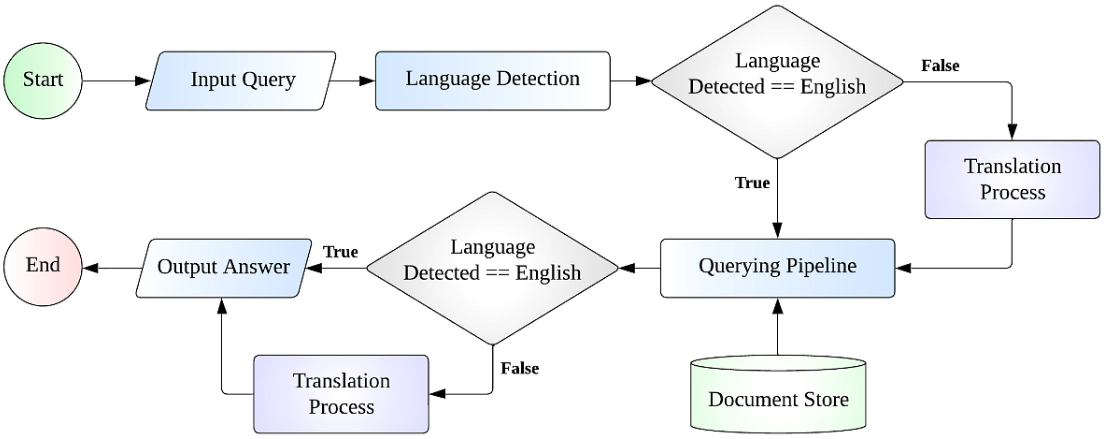
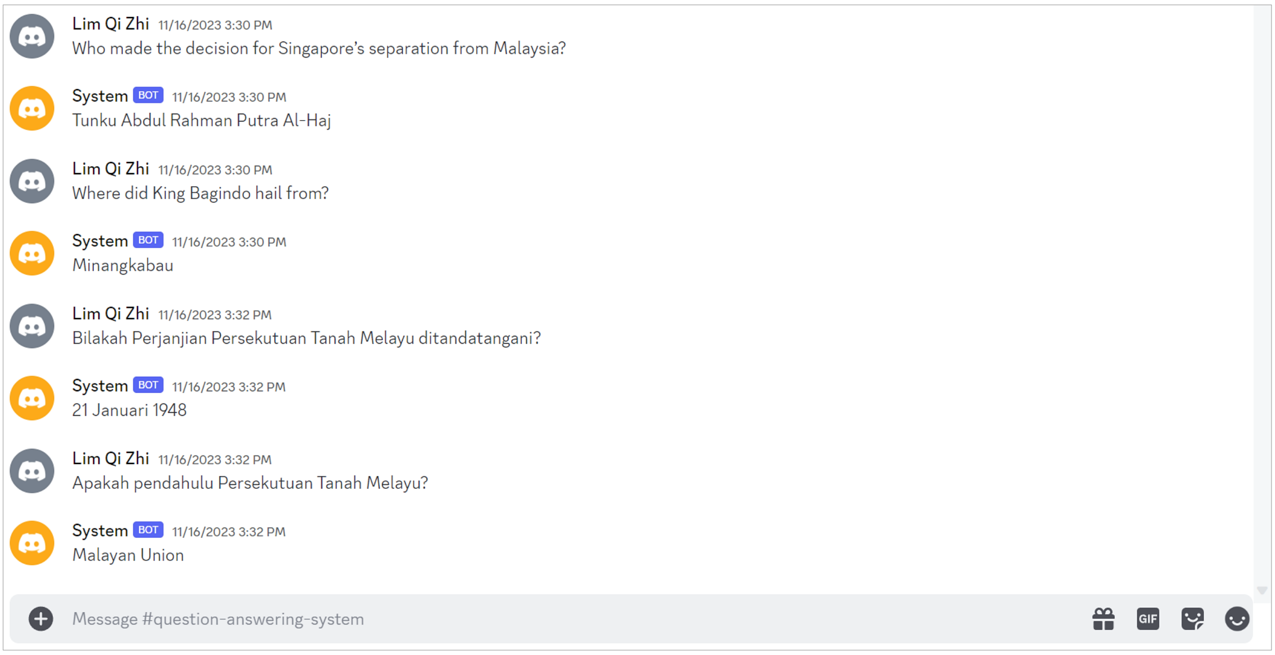


Figure 2. Flowchart of the Deployed QAS in Discord

The process begins when the user inputs a query into the system, which subsequently detects the language of the query. The proposed system is designed to accommodate both Malay and English. If the language detected is Malay, translation becomes necessary as the data used is in English. Conversely, if the language detected is English, translation is not required. The query is then forwarded to the querying pipeline, which functions to retrieve relevant context and extract the best answer to the query. It is important to note that the answer returned will be converted back to the language of the original query if needed. Finally, the output answer is presented to the user.

Figure 3 displays a screenshot of the proposed Multilingual QAS for Malaysia History deployed in Discord, showcasing its capability in effectively answering questions related to the history of Malaysia. The examples provided in the screenshot also demonstrate that the system is equipped with multilingual support, enabling it to answer questions from users in both English and Malay.



Lim Qi Zhi 11/16/2023 3:30 PM
Who made the decision for Singapore's separation from Malaysia?

System BOT 11/16/2023 3:30 PM
Tunku Abdul Rahman Putra Al-Haj

Lim Qi Zhi 11/16/2023 3:30 PM
Where did King Bagindo hail from?

System BOT 11/16/2023 3:30 PM
Minangkabau

Lim Qi Zhi 11/16/2023 3:32 PM
Bilakah Perjanjian Persekutuan Tanah Melayu ditandatangani?

System BOT 11/16/2023 3:32 PM
21 Januari 1948

Lim Qi Zhi 11/16/2023 3:32 PM
Apakah pendahulu Persekutuan Tanah Melayu?

System BOT 11/16/2023 3:32 PM
Malayan Union

Figure 3. Screenshot of the Multilingual QAS for Malaysia History

6- Conclusion

In conclusion, this study has successfully developed a Multilingual Question Answering System (QAS) for Malaysia History by utilizing a transformer-based language model. It serves as a proof of concept, demonstrating the effectiveness of transformer-based language models in answering historical questions. The system development process involved several major steps, including data collection from the Internet, knowledge representation in separate text files, data translation from Malay to English, document loading and pre-processing, document indexing and storing into DocumentStore, and the initialization of the querying pipeline with retriever and reader modules to retrieve relevant context and extract answers for user queries.

In order to determine the optimal reader model, a comparative analysis of six different transformer-based language models, including DeBERTaV3, BERT, ALBERT, ELECTRA, MiniLM, and RoBERTa, has been conducted. The performance of the models was evaluated using a dataset comprising 100 question-answer pairs related to the history of Malaysia, which was meticulously created for this study. Four evaluation metrics, namely average F1-score, average Jaccard Similarity, average cosine similarity, and average semantic similarity, were employed to assess the performance of the proposed QAS. As a result, the RoBERTa model stands out as the best performer, demonstrating the best overall performance on the question-answering task.

Based on the experimental results, RoBERTa was employed as the reader model for the proposed QAS to effectively extract answers for user queries. Lastly, the proposed QAS was deployed on Discord, which is a widely used communication platform. By incorporating language detector and translator modules, the proposed system is capable of handling multilingual queries from users. In this case, users can ask questions in either Malay or English, and they will receive responses in the original language they used.

In the future development of multilingual QAS, it is worthwhile to investigate the utilization of multilingual models for more seamless and enhanced multilingual support. Additionally, fine-tuning of the transformer-based language models can be adopted to further improve their performance for specific use cases. Furthermore, exploring the potential of generative QAS is also worth considering. Instead of solely highlighting the specific span of text that answers a query, a generative QAS can generate a novel textual response for the user. This opens up possibilities for generating more creative and diverse answers for the users in the question-answering process.

7- Declarations

7-1- Author Contributions

Conceptualization, Q.Z.L., J.X.N., E.K.H.O., N.K.N.L., C.P.L. and K.M.L.; methodology, Q.Z.L., J.X.N., E.K.H.O., N.K.N.L., C.P.L. and K.M.L.; software, Q.Z.L., J.X.N., E.K.H.O. and N.K.N.L.; validation, Q.Z.L., J.X.N., E.K.H.O., N.K.N.L., C.P.L. and K.M.L.; formal analysis, C.P.L. and K.M.L.; investigation, Q.Z.L., J.X.N., E.K.H.O. and N.K.N.L.; resources, Q.Z.L. and J.X.N.; data curation, Q.Z.L., J.X.N., E.K.H.O., N.K.N.L., C.P.L. and K.M.L.; writing—original draft preparation, Q.Z.L., J.X.N., E.K.H.O., N.K.N.L., C.P.L., and K.M.L.; writing—review and editing, Q.Z.L., J.X.N., E.K.H.O., N.K.N.L., C.P.L. and K.M.L.; visualization, Q.Z.L., J.X.N., E.K.H.O. and N.K.N.L.; supervision, C.P.L. and K.M.L.; project administration, C.P.L. and K.M.L.; funding acquisition, C.P.L. and K.M.L. All authors have read and agreed to the published version of the manuscript.

7-2- Data Availability Statement

The data presented in this study are available on request from the corresponding author.

7-3- Funding

This research was funded by Telekom Malaysia R&D under grant number RDTC/221064 and RDTC/231075.

7-4- Institutional Review Board Statement

Not applicable.

7-5- Informed Consent Statement

Not applicable.

7-6- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

8- References

[1] Chee-Huay, C., & Kee-Jiar, Y. (2016). Why Students Fail in History: A Minor Case Study in Malaysia and Solutions from Cognitive Psychology Perspective. Mediterranean Journal of Social Sciences. doi:10.5901/mjss.2016.v7n1p517.

[2] Woods, W., Kaplan, R. M., & Nash-Webber, B. L. (1972). The Lunar Science Natural Language Information System: Final Report. BBN Report No. 11501, Contract No. NAS9-1115 NASA Manned Spacecraft Center, Houston, Texas, United States.

[3] Androutsopoulos, I., Ritchie, G. D., & Thanisch, P. (1996). A Framework for Natural Language Interfaces to Temporal Databases. Proceedings of the 20th Australasian Computer Science Conference, 5–7 February, 1997, Sydney, Australia.

[4] Ojokoh, B., & Adebisi, E. (2019). A review of question answering systems. Journal of Web Engineering, 17(8), 717–758. doi:10.13052/jwe1540-9589.1785.

[5] Zheng, Z. (2002). AnswerBus question answering system. Proceedings of the Second International Conference on Human Language Technology Research, 399-404. doi:10.3115/1289189.1289238.

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. 31st Conference on Neural Information Processing Systems (NIPS 2017), 4-9 December, 2017, Long Beach, United States.

[7] Ou, Y.-Y., Chuang, S.-W., Wang, W.-C., & Wang, J.-F. (2022). Automatic Multimedia-based Question-Answer Pairs Generation in Computer Assisted Healthy Education System. 2022 10th International Conference on Orange Technology (ICOT), Shanghai, China. doi:10.1109/icot56925.2022.10008119.

[8] Zhang, J. (2022). Application Research of Similarity Algorithm in the Design of English Intelligent Question Answering System. 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), Karnataka, India. doi:10.1109/icmnwc56175.2022.10031708.

[9] Das, B., & Nirmala, S. J. (2022). Improving Healthcare Question Answering System by Identifying Suitable Answers. 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India. doi:10.1109/mysurucon55714.2022.9972435.

[10] Gupta, S. (2023). Top K Relevant Passage Retrieval for Biomedical Question Answering. arXiv preprint arXiv:2308.04028. doi:10.48550/arXiv.2308.04028.

[11] Pudasaini, S., & Shakya, S. (2023). Question Answering on Biomedical Research Papers using Transfer Learning on BERT-Base Models. 2023 7th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Kirtipur, Nepal. doi:10.1109/i-smac58438.2023.10290240.

[12] Alzubi, J. A., Jain, R., Singh, A., Parwekar, P., & Gupta, M. (2023). COBERT: COVID-19 Question Answering System Using BERT. Arabian Journal for Science and Engineering, 48(8), 11003–11013. doi:10.1007/s13369-021-05810-5.

[13] Acharya, S., Sornalakshmi, K., Paul, B., & Singh, A. (2022). Question Answering System using NLP and BERT. 3rd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India. doi:10.1109/icosec54921.2022.9952050.

[14] Yin, J. (2022). Research on Question Answering System Based on BERT Model. 2022 3rd International Conference on Computer Vision, Image and Deep Learning &amp; International Conference on Computer Engineering and Applications (CVIDL & ICCEA), Changchun, China. doi:10.1109/cvidliccea56201.2022.9824408.

[15] Yang, J., Yang, X., Li, R., Luo, M., Jiang, S., Zhang, Y., & Wang, D. (2023). BERT and hierarchical cross attention-based question answering over bridge inspection knowledge graph. Expert Systems with Applications, 233, 120896. doi:10.1016/j.eswa.2023.120896.

[16] Tian, D., Li, M., Ren, Q., Zhang, X., Han, S., & Shen, Y. (2023). Intelligent question answering method for construction safety hazard knowledge based on deep semantic mining. Automation in Construction, 145, 104670. doi:10.1016/j.autcon.2022.104670.

[17] Liu, S., & Huang, X. (2019). A Chinese Question Answering System based on GPT. 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China. doi:10.1109/icsess47205.2019.9040807.

[18] Noraset, T., Lowphansirikul, L., & Tuarob, S. (2021). WabiQA: A Wikipedia-Based Thai Question-Answering System. Information Processing & Management, 58(1), 102431. doi:10.1016/j.ipm.2020.102431.

[19] Ainon, R. N., Salim, S. S., & Noor, N. E. M. (1989). A question-answering system in Bahasa Malaysia. Fourth IEEE Region 10 International Conference TENCON, Bombay, India. doi:10.1109/tencon.1989.176892.

[20] Puteh, N., Husin, M. Z., Tahir, H. M., & Hussain, A. (2019). Building a question classification model for a Malay question answering system. International Journal of Innovative Technology and Exploring Engineering, 8(5s), 184–190.

[21] Lim, H. T., Huspi, S. H., & Ibrahim, R. (2021). A Conceptual Framework for Malay-English Mixed-language Question Answering System. 2021 International Congress of Advanced Technology and Engineering (ICOTEN), Taiz, Yemen. doi:10.1109/icoten52080.2021.9493503.

[22] Pietsch, M., Möller, T., Kostic, B., Risch, J., Pippi, M., Jobanputra, M., Zanzottera, S., Cerza, S., Blagojevic, V., Stadelmann, T., Soni, T., & Lee, S. (2019). Haystack: the end-to-end NLP framework for pragmatic builders. Available online: https://github.com/deepset-ai/haystack (accessed on March 2024).

[23] Pietsch, M., Möller, T., Kostic, B., Risch, J., Pippi, M., Jobanputra, M., ... & Lee, S. (2019). Haystack: the end-to-end NLP framework for pragmatic builders.

[24] Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., & Ahmed, A. (2020). Big bird: Transformers for longer sequences. 34th Conference on Neural Information Processing Systems (NeurIPS 2020), 6-12 December, 2020, Vancouver, Canada.

[25] Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don't know: Unanswerable questions for SQuAD. arXiv preprint arXiv:1806.03822. doi:10.18653/v1/p18-2124.

[26] He, P., Gao, J., & Chen, W. (2021). Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. arXiv preprint arXiv:2111.09543.

[27] He, P., Liu, X., Gao, J., & Chen, W. (2020). Deberta: Decoding-enhanced BERT with disentangled attention. arXiv preprint arXiv:2006.03654. doi:10.48550/arXiv.2111.09543.

[28] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. doi:10.48550/arXiv.1810.04805.

[29] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite BERT for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942. doi:10.48550/arViv.1909.11942.

[30] Clark, K., Luong, M. T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:2003.10555. doi:10.48550/arXiv.2003/10555.

[31] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). MINILM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. 34th Conference on Neural Information Processing Systems (NeurIPS 2020), 6-12 December, 2020, Vancouver, Canada.

[32] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692. doi:10.48550/arXiv.1907.11692.