ADBA Chaos and Fractals
Cutting-Edge Scientific Solutions

e-ISSN: 3023-8951
RESEARCH ARTICLE
Vol.1 / No.1 / 2024 / pp.19-30
https://doi.org/10.69882/adba.chf.2024073

Comparative Analysis of State-of-the-Art Q&A Models: BERT, RoBERTa, DistilBERT, and ALBERT on SQuAD v2 Dataset

Cem Özkurt *,[1]
*Sakarya University of Applied Sciences, Artificial Intelligence and Data Science Application and Research Center, 54050, Sakarya, Türkiye.

ABSTRACT In the rapidly evolving landscape of natural language processing (NLP) and artificial intelligence, recent years have witnessed significant advancements, particularly in text-based question-answering (QA) systems. The Stanford Question Answering Dataset (SQuAD v2) has emerged as a prominent benchmark, offering diverse language understanding challenges. This study conducts a thorough examination of cutting-edge QA models—BERT, DistilBERT, RoBERTa, and ALBERT—each featuring distinct architectures, focusing on their training and performance on SQuAD v2. The analysis aims to uncover the unique strengths of each model, providing insights into their capabilities and exploring the impact of different training techniques on their performance. The primary objective is to enhance our understanding of text-based QA systems' evolution and their effectiveness in real-world scenarios. The results of this comparative study are poised to influence the utilization and development of these models in both industry and research. The investigation meticulously evaluates BERT, ALBERT, RoBERTa, and DistilBERT QA models using the SQuAD v2 dataset, emphasizing instances of accurate responses and identifying areas where completeness may be lacking. This nuanced exploration contributes to the ongoing discourse on the advancement of text-based question-answering systems, shedding light on the strengths and limitations of each QA model. Based on the results obtained, ALBERT achieved an exact match of 86.85% and an F1 score of 89.91% on the SQuAD v2 dataset, demonstrating superior performance in both answerable ('HasAns') and unanswerable ('NoAns') questions. BERT and RoBERTa also showed strong performance, while DistilBERT lagged slightly behind. This study provides a significant contribution to the advancement of text-based question-answering systems, offering insights that can shape the utilization of these models in both industry and research domains.

KEYWORDS
Question-Answering models
BERT
RoBERTa
DistilBERT
ALBERT
SQuAD v2
Dataset
Model performance

INTRODUCTION

Recent years have witnessed significant advancements in natural language processing (NLP) and artificial intelligence, particularly in text-based question-answering (QA) systems. Research in this field aims to develop new methods and approaches to enhance the effectiveness of QA systems used in various domains. These developments have led to the emergence of benchmarks such as the Stanford Question Answering Dataset (SQuAD v2) and facilitated the comparison of QA model performances. For instance, one such study introduces "RealTime QA," a dynamic QA platform that poses questions about current world events and regularly evaluates systems. This platform challenges the static assumptions present in traditional open-domain QA datasets and emphasizes real-time applications. The models developed serve as strong baseline models built upon large pretrained language models, thus making significant strides in real-time QA services (Kasai *et al.* 2024).

Additionally, a system named "Visconde" proposes a solution for answering questions that require evidence spread across multiple documents. This system employs a three-step pipeline to address the task, highlighting that current retrievers are often the primary bottleneck and that models perform at human levels given relevant passages (Pereira *et al.* 2023).

Furthermore, a review on how transformer models are applied in text-based QA systems sheds light on recent trends in the field. This review discusses different transformer architectures, attention mechanisms, and evaluation metrics used to assess the performance of QA systems (Nassiri and Akhloufi 2023). Moreover, a study addressing the information-intensive construction industry

Manuscript received: 25 June 2024,
Revised: 10 July 2024,
Accepted: 11 July 2024.

[1]cemozkurt@subu.edu.tr (**Corresponding author**)

develops a query-answering (QA) system using natural language processing (NLP) methods. This system aims to support decision-making processes in construction projects by creating virtual assistants (Wang *et al.* 2022). The contributions of these studies not only aid in understanding and improving current text-based QA applications but also lay a foundation for future research endeavors. These advancements have the potential to make QA systems more effective and usable in both industry and academia (Caballero 2021).

In recent years, Natural Language Processing (NLP) has undergone remarkable advancements, primarily attributed to the transformative influence of the Transformer architecture and the selfattention mechanism. These attention-based models have showcased unparalleled performance across various NLP benchmarks, fueled in part by the growing popularity of transfer learning. This section provides an overview of related works that delve into the application and comparative analysis of Transformerbased models across different domains.

Rawat and Samant (2022) conducted a comparative analysis of transformer-based models for question answering, delving into models such as BERT, ALBERT, RoBERTa, XLNET, DistilBERT, Electra, and Pegasus. Their study, centered on Question-Answering (QA) systems using the SQUAD2 dataset, emphasizes the evolution from traditional "Bag of Words" methods to more efficient transformer libraries, exemplified by HuggingFace's BERT Question Answering model. This approach significantly enhances the models' capability to answer questions from large documents. Nassiri and Akhloufi (2023) contribute with a comprehensive review of studies focusing on the use of transformer models in text-based question-answering systems. The paper categorizes transformer architectures based on encoders, decoders, and encoder-decoder structures. It explores recent trends in textual QA datasets, providing insights into QA system architectures and evaluation metrics. The authors underscore the need for simplified implementation of Transformer models.

Kumari *et al.* (2022b) present a comparative analysis of transformer-tased models for document visual question answering, concentrating on Visual Question Answering (VQA), specifically the DocVQA task. The study investigates transformer models such as BERT, ALBERT, RoBERTa, ELECTRA, and Distil-BERT. The analysis includes a detailed examination of validation accuracy, considering challenges posed by documents, layout understanding, and writing patterns. Sabharwal and Agrawal (2021) explore the intricacies of the BERT algorithm for sentence embedding and various training strategies, providing a practical application in text classification systems. This chapter serves as a valuable resource for understanding BERT's applications in Neural Networks and Natural Language Processing.

Gillioz *et al.* (2020) offer an overview of transformer-based models for NLP tasks, discussing the transformative impact of the transformer architecture on NLP since its proposal in 2017. The authors cover auto-regressive models like GPT, GPT-2, and XLNET, as well as auto-encoder architectures like BERT and post-BERT models, including RoBERTa, ALBERT, ERNIE 1.0/2.0. Sidorov *et al.* (2023) analyze the performance of different transformer models for regret and hope speech detection, highlighting their effectiveness and superiority in regret detection. The study emphasizes the importance of considering specific transformer architectures and pre-training for different tasks.

Pirozelli *et al.* (2022) propose an innovative approach to QA systems, exploring dual system architectures that filter unanswerable or meaningless questions. The paper presents experiments using classification and regression models to filter questions, demonstrating that this modular approach contributes to improving the quality of answers generated by QA systems. Nassiri and Akhloufi (2023) delve into the application of transformer models in text-based questionanswering systems, emphasizing their significance in natural language processing (NLP). The study provides a comprehensive review, categorizing transformer architectures based on encoders, decoders, and encoder-decoder structures. The authors also highlight recent trends in textual QA datasets, 2 system architectures, and evaluation metrics, underscoring the need for simplified implementation of transformer models.

Ghanem *et al.* (2023) tackle the issue of spam on social networks by proposing a RoBERTabased bi-directional Recurrent Neural Network for spam detection. Their study demonstrates superior performance, outperforming common transformer-based models on benchmark datasets from Twitter, YouTube, and SMS. MacRae (2022) details the development and deployment of NOLEdge, an intelligent search tool for the Florida State University Computer Science department. The study involves fine-tuning a pretrained transformer model and explores various methods of textual data augmentation, contributing insights into the model's efficacy and potential areas for further research.

Tahsin Mayeesha *et al.* (2021) address the gap in Bengali language processing, focusing on training transformer models for question answering. The study utilizes synthetic reading comprehension datasets and human-annotated Bengali QA datasets, comparing the models with human performance through survey experiments. Schütz *et al.* (2021) propose a content-based classification approach for automatic fake news detection using various pre-trained transformer models. The study reveals the effectiveness of transformers in achieving high accuracy on the FakeNewsNet dataset, emphasizing their potential impact on combating misinformation.

Kumari *et al.* (2022a) contribute to the field of question answering and generation, introducing novel transformer-based models like BERT, AIBERT, and DistilBERT. Their work integrates question generation with question answering systems, showcasing the models' capabilities in suggesting relevant questions based on input context. David (2020) explores the representation learning of autoencoding transformer models in ad hoc information retrieval, evaluating various transformer architectures such as BERT, RoBERTa, and DistilBERT. The study provides insights into the performance of these models in tasks like semantic similarity and their suitability for ad hoc document retrieval.

Malla and Alphonse (2021) address the COVID-19 outbreak, developing an ensemble pre-trained deep learning model for detecting informative tweets. Their Majority Voting technique-based Ensemble Deep Learning (MVEDL) model demonstrates high accuracy in identifying COVID-19 related informative tweets. Srivastava *et al.* (2021) investigate brand perception in marketing, probing contextual language models, including BERT and GPT, for associations with brand attributes. The study aims to understand the encoded brand perceptions and their potential impact on downstream tasks. Greco *et al.* (2022) provide a comprehensive comparison of transformer-based language models on NLP benchmarks, shedding light on the strengths and weaknesses of various models in different NLP tasks.

Sundelin (2023) explores the use of transformer models in identifying toxic language online, comparing the performance of RoBERTa, ALBERT, and DistilBERT. The study reveals distinctions in their efficiency based on datasets and real-world evaluations. Kumar *et al.* (2023) investigate modern question-answering ma-

chines, focusing on BERT and its variants. The study outlines the operation of machine reading comprehension and its application in providing in-depth solutions to user queries.

In conclusion, these collective works significantly contribute to the understanding of transformer models' applications across diverse domains, showcasing their effectiveness in addressing complex challenges in natural language processing and information retrieval.

MATERIALS AND METHODS

One of the significant strengths of this study lies in its comprehensive comparative analysis of four different BERT-based models (DistilBERT, BERT, ALBERT, and RoBERTa) on the SQuAD v2 dataset. Unlike many other studies that focus on a single model or dataset, this research provides a detailed evaluation across multiple models, offering a broader perspective on their relative performances. Additionally, our work uniquely emphasizes the practical implications of model performance in specific scenarios. By considering computational resource limitations, we highlight DistilBERT's value, making this study highly relevant for applications with constrained resources. The inclusion of a thorough discussion on hyperparameter optimization and its impact on model performance further distinguishes our research. This attention to fine-tuning details provides actionable insights for practitioners looking to maximize the efficiency of BERT-based models. Furthermore, our analysis includes both answerable and unanswerable questions, offering a more nuanced understanding of each model's strengths and weaknesses. This dual focus enhances the study's applicability in real-world settings, where the ability to handle unanswerable questions is critical. Overall, the comprehensive approach, practical relevance, and detailed methodological insights contribute to making this study a valuable resource for the NLP community.

BERT and Its Impact

The advent of Bidirectional Encoder Representations from Transformers (BERT) by Devlin *et al.* (2018) has revolutionized the field of Natural Language Processing (NLP). BERT is a deep learning model designed to understand the context of a word in search queries, making it highly effective for tasks like question answering, language inference, and others.

BERT's transformer architecture allows it to achieve state-of-the-art results by pre-training on a large corpus and fine-tuning on specific tasks. The model employs a bidirectional approach, considering the context from both directions, which sets it apart from traditional models like LSTM and GRU. Studies by Liu *et al.* (2019); Yang *et al.* (2019) have shown BERT's superior performance in various NLP benchmarks.

Application in Question Answering

The SQuAD (Stanford Question Answering Dataset) has been a benchmark for evaluating question-answering models. The release of SQuAD v2 introduced unanswerable questions, adding a new challenge for models to determine when no answer is possible. Prior works such as by Lan *et al.* (2019) have addressed these challenges by enhancing model architectures and training strategies.

Our study builds on these foundations, aiming to fine-tune BERT on the SQuAD v2 dataset to push the boundaries of question-answering capabilities. By focusing on the nuances of unanswerable questions and optimizing hyperparameters, we seek to achieve a new benchmark in performance.

Datasets

For the training and evaluation of our question-answering systems, we employed the widely recognized "Squad v2" dataset, developed by Stanford University specifically for natural language processing (NLP) tasks. Serving as an enhanced iteration of "Squad v1," this dataset comprises over 100,000 instances, each consisting of a paragraph and a corresponding question.

The "Squad v2" dataset Yu and Sun (2023) is partitioned into two subsets: the training set, featuring 130,319 question-answer pairs, and the validation set, with 11,873 pairs. A key attribute of the dataset is the equitable distribution of answerable and unanswerable questions in both sets, ensuring a balanced evaluation approach. Stored in JSON format, each dataset file includes an array of data items. These items consist of a title and a set of paragraphs, wherein each paragraph contains text and a list of associated questions. Each question is uniquely identified by an ID, question text, answerability label, and one or more answers, each specified by text and character position. To quantify the linguistic complexity, the text in the "Squad v2" dataset undergoes tokenization, resulting in a total of 1,535,809 tokens. The training set encompasses 1,321,104 tokens, while the validation set comprises 214,705 tokens. Notably, the tokenization method segments text based on word boundaries and punctuation marks, eliminating spaces.

A distinctive feature of "Squad v2" is the incorporation of "impossible" instances, strategically designed to enhance accuracy measurement by introducing challenging scenarios. These instances include situations where answers are not present in the text or where questions contain incomplete or misleading information. The dataset's comprehensive coverage of diverse topics and real-world meaning contributes to the improvement of model generalization capabilities. Through the strategic utilization of the "Squad v2" dataset, this study aims to provide a robust and comprehensive performance measurement grounded in real-world scenarios, both during the training phase and result assessment.

Within Figure 1, we encounter a screenshot showcasing a question and its corresponding answer from the SQuAD v2 dataset. This dataset is a widely utilized question-answer dataset in natural language processing (NLP) research, composed of text and associated queries. The presented question in Figure 1 is, 'How did Marco Polo acquire information about China?' This question has sparked numerous debates due to the lack of concrete evidence regarding Marco Polo's visit to China. Some argue that Polo obtained information through contact with Persian traders. The answer provided is 'Through contact with Persian traders.' This answer is extracted from the following paragraph in the text: 'Marco Polo acquired much of his knowledge through contact with Persian traders since many of the places he named were in Persian.' This paragraph states that many of the places Polo observed in China had Persian names, serving as evidence of his visit. However, this evidence is not conclusive, and Polo could have acquired information about China from Persian traders. Figure 1 serves as a crucial piece of evidence supporting the main idea of our article. It highlights the necessity for NLP models to possess the capability to comprehend and analyze text for effectively answering complex and open-ended questions. In Figure 1, we observe a question and its answer from the SQuAD v2 dataset. The question explores how Marco Polo acquired knowledge about China, with the answer being through contact with Persian traders. This answer is derived from the following paragraph in the text: 'Marco Polo acquired much of his knowledge through contact with Persian traders since many of the places he named were in Persian.' This paragraph

Figure 1 Squad v2 Datasets Example

notes that the locations Polo observed in China often had Persian names, considered evidence of his visit. However, this evidence is inconclusive, leaving the possibility that Polo obtained information about China from Persian traders. This evidence emphasizes that NLP models must possess the ability to comprehend and analyze text for answering complex and open-ended questions effectively. Such questions demand the integration of information from different sections of the text, highlighting the importance of understanding and analyzing textual content. This evidence strongly supports the main idea of our article, which argues that NLP models must possess the capability to comprehend and analyze text for effectively answering complex and open-ended questions.

Model Architecture

BERT: BERT (Bidirectional Encoder Representations from Transformers) Fu *et al.* (2023) stands out as a pivotal achievement in natural language processing (NLP), having demonstrated considerable success. In the initial pre-training phase, the model acquires a broad understanding of language structure and context through unsupervised learning on extensive text corpora. Fine-tuning for specific tasks, such as question-answering (QA), further refines its capabilities, enabling accurate responses to posed questions within given text passages. Utilizing tokens like [CLS] (classification) and [SEP](separator) to structure input text in QA tasks, BERT's token embedding layers transform each word and token into numerical representations, enhancing overall understanding. The selfattention mechanism, inherent in its bidirectional attention design, amalgamates information from all words in the text to grasp the context of each word. Typically, outputs are derived from the [CLS] token and used for specific classification tasks, such as predicting the appropriate answer class for a given question. BERT's structured transformer network renders it adaptable to various NLP applications, showcasing particularly effective performance in question-answering tasks.

In this study, the research expands upon BERT's capabilities, focusing on its application in question-answering tasks. The model, pre-trained for the general QA task, is fine-tuned using the Squad v2 dataset. Similar to DistilBERT, BERT integrates both the question and context, drawing upon language representations obtained during pre-training to comprehend the information and generate precise answers to posed questions. The comprehensive examination of BERT's performance in the QA model, conducted with the Squad v2 dataset, is detailed within the Model Architecture section under the Methodology. This inclusion aims to elucidate BERT's distinctive role and capabilities in the context of this research.The steps and mathematical formula of the BERT model are shown in Figure 2.

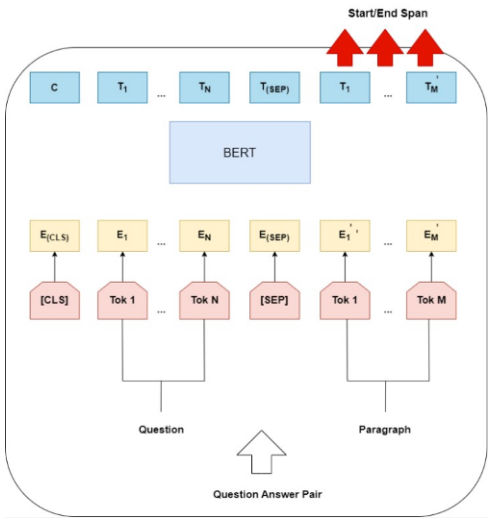

Figure 2 Use of BERT for question and answer

Figure 2 illustrates the functioning of the BERT QA model. The model is trained with a questionanswer pair consisting of a question designed to be posed by a human and a corresponding paragraph. The question is formulated in a way that seeks information, and the answer is based on the information present in the paragraph.

The model operates in two stages:

• In the first stage, the model converts the question and paragraph into a sequence of tokens. Tokens are the basic language elements that the model can comprehend. For instance, changing the question to "What is the capital of Turkey?" would result in tokens such as "Turkey," "capital," "is," and "?". • In the second stage, the model processes the tokens through a BERT model. BERT is a deep learning model with the ability to understand the context of a word in a sentence. The model is used to determine the meanings

of tokens and their relationships.

The text in the image defines different token types used by the model:

• Start/End Span tokens specify the start and end positions of the answer token in the paragraph. For example, for the question "What is the capital of Turkey?" the word "Ankara" could be an answer token, and the Start/End Span tokens would indicate its position in the paragraph.

• The CLS token indicates the question token. For instance, for the question "What is the capital of Turkey?" the CLS token would precede the word "Turkey."

• N tokens denote normal word tokens in the paragraph. For the question "What is the capital of Turkey?" the word "capital" would be an N token.

• M tokens denote stop-word tokens in the paragraph. For example, the "?" sign is an M token.

After processing the tokens, the model produces an answer token that can address the question. In this case, the answer token would be generated as "Ankara" based on the information present in the paragraph, specifying the position of "Ankara" with Start/End Span tokens.

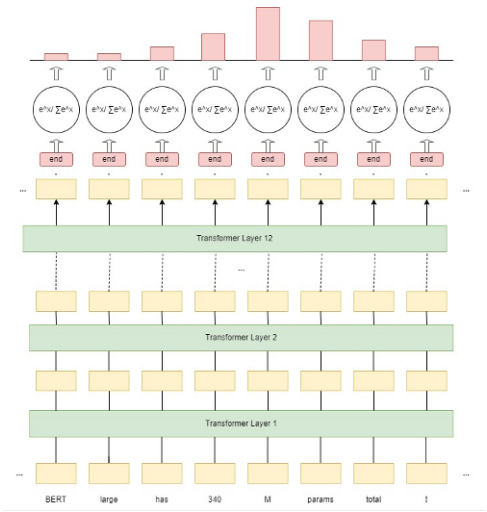

Figure 3 Layers passed in the Bert QA model

Figure 3 depicts the block diagram of the BERT QA model, revealing its three-tiered architecture. It illustrates the three layers that constitute the BERT model. The bottommost layer is the input layer, representing textual data. The middle layer serves as a transformative layer, employed to discern relationships among textual data. The top layer functions as an output layer, crucial for responding to queries.

• Input Layer : It is a vector representing textual data, encapsulating either a summary of the text document or the text question itself.

• Transformative Layer : This layer is dedicated to understanding the interconnections among textual data. It is intricately divided into twelve layers, interlinked through a pair of attention mechanisms.

• Attention Mechanisms : These mechanisms are utilized to ascertain the relationship of a specific point in the text with other textual segments. This aids the BERT model in comprehending the meaning of textual data.

• Output Layer : This layer consists of a vector used to respond to queries. This vector encapsulates either the answer to the question or a summary related to the answer.

This innovative depiction showcases the intricate design of the BERT QA model, emphasizing its capacity to unravel the complexities of textual relationships through attention mechanisms, ultimately providing insightful responses to posed questions.

Figure 4 depicts the internal structure of the Encoder and Decoder components of the BERT architecture.

The Encoder takes input from text or code and transforms it into a sequence of vectors. These vectors represent the meaning of the text and the position in the sentence. The Encoder consists of three main layers:

• Embedding Layer : This layer transforms each word into a vector.

• Multi-head Attention Layer : This layer learns relationships between words.

• Positional Encoding Layer : This layer represents the position of words in the sentence.

The Decoder takes vectors from the Encoder and produces the output. Similar to the Encoder, the Decoder comprises three main layers:

• Embedding Layer : This layer transforms each word into a vector.

• Multi-Head Attention Layer : This layer learns relationships between words.

• Positional Encoding Layer : This layer represents the position of words in the sentence.

Both Encoder and Decoder layers utilize a technique called attention mechanism. The attention mechanism is used to determine the relationship between a word and other words. This enables BERT to understand relationships between texts.

Fig.4 provides a detailed view of the internal structure of the Encoder and Decoder components. This information can help you better understand how the BERT architecture operates.

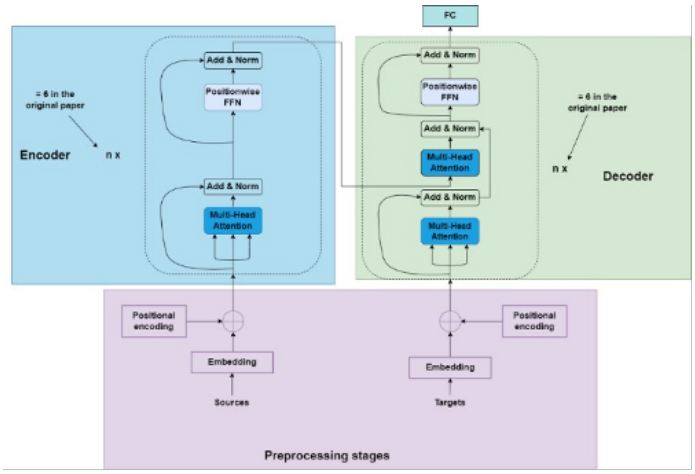

Figure 4 BERT QA Model Encoder-Decoder-Preprocessing Stages Relationship

Figure 4 illustrates the intricate relationship among the Encoder, Decoder, and Preprocessing Stages within the BERT architecture. This symbiotic connection is crucial for the effective functioning of BERT-based Question Answering (QA) models. Let's delve into the nuanced interplay between these components. The Preprocessing Stages serve as the initial gateway for input data. This phase involves tokenizing the text or code, converting words into vectors through the Embedding layer, and incorporating positional encoding to capture the contextual information of each word in the

sentence. The transformed input from the Preprocessing Stages then proceeds to the Encoder. The Embedding layer of the Encoder is pivotal in converting each tokenized word into a vector, providing a foundational representation of the input. The Multi-head attention layer follows suit, fostering an understanding of relationships between words. Simultaneously, the Positional encoding layer imparts valuable information regarding the spatial arrangement of words within the sentence. Moving on to the Decoder, it receives the enriched vectors from the Encoder and embarks on a similar journey. The Embedding layer refines the representations of words, while the Multi-head attention layer delves into the intricate web of relationships between words. The Positional encoding layer ensures that the positional information is retained throughout this process. The crux of this dynamic lies in the attention mechanism, a common thread woven into both the Encoder and Decoder. This mechanism empowers BERT to discern the significance of each word in relation to others, facilitating a holistic understanding of contextual dependencies.

In summary, the Preprocessing Stages lay the groundwork by transforming input data, the Encoder refines and enriches these representations, and the Decoder further refines them while decoding the final output. The cohesive interplay between these stages, guided by the attention mechanism, equips BERT-based QA models with the ability to grasp intricate relationships and nuances within the given text or code. This elucidation provides a concise yet comprehensive understanding of the intricate relationship between the Encoder, Decoder, and Preprocessing Stages as depicted in Figure 4.

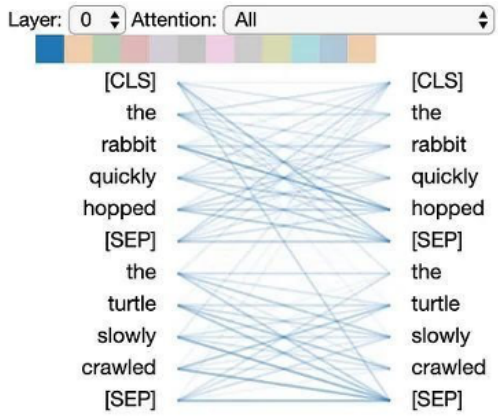

Figure 5 BERT Input Format and Textual

This textual example, [CLS] the rabbit quickly hopped [SEP] the turtle slowly crawled [SEP], can be elucidated as follows:

This instance represents the specific input format of the BERT model. The [CLS] token signifies the beginning and typically acts as a representative of the query. [SEP] tokens are used to separate two distinct text segments. The first [SEP] indicates the first text segment and the subsequent group of words. The second [SEP] denotes the second text segment and the following group of words.

In this example, the phrase "the rabbit quickly hopped" represents one text segment, while "the turtle slowly crawled" represents a second text segment. The differing relationships between them are based on the content of these two text segments. The occurrence of the rabbit swiftly hopping in the first segment implies one event, whereas the turtle slowly crawling in the second segment conveys a different event. BERT utilizes a learned attention mechanism to comprehend such relationships and grasp the context of text segments. This format serves as an example to showcase BERT's ability to understand relationships within text. When processing such inputs, the model can compare different text segments and decipher relationships. This explanation provides insight into the Figure 5, highlighting how BERT processes input data with distinct text segments, drawing attention to its learned attention mechanism for understanding textual relationships.

DistilBERT: DistilBERT is a distilled version of the BERT (Bidirectional Encoder Representations from Transformers) model, designed for natural language processing tasks such as Question Answering (QA). This model has been pre-trained to adapt to a customized QA task. DistilBERT is often characterized by smaller dimensions and parameters, designed to be smaller and faster than the original BERT model. However, this design choice may result in a slight loss of information compared to the original BERT model. Nevertheless, it offers advantages, especially in resource-constrained environments. Furthermore, it preserves the general learning capabilities of the original BERT model (Benedetto 2023).

In the QA task, the DistilBERT model focuses on providing answers to specific questions over a text paragraph or document. The model combines the question and context, utilizes the language representations it learned during pre-training to comprehend this information, and then generates an appropriate answer to the question. A pre-trained DistilBERT model, having general language understanding capabilities, can be fine-tuned for a specific QA task to be customized for better performance on a particular topic or dataset.

In this study, we trained DistilBERT with the Squad v2 dataset and describe its performance on the QA model. The introduction of DistilBERT within the Model Architecture section under the Methodology aims to provide a comprehensive understanding of its role and capabilities in the context of this research.

Figure 6's diagram elucidates the operational dynamics of the DistilBERT QA Model. The model processes inputs in two stages. In the first stage, inputs undergo tokenization and pass through an embedding layer, constructing a vector representation of the inputs. In the second stage, leveraging the vector representation of inputs, the model executes a question-answering task employing a series of transformer layers and a prediction layer.

• Inputs and Outputs : The model's inputs consist of a question and a context text. The question represents what we seek to know the answer to, while the context text embodies the text containing the answer to the question. The model's output is the answer to the question.

• Embedding Layer : This layer tokenizes inputs and generates a vector representation for each token, culminating in an overall vector representation of the inputs.

• Transformer Layers : Utilizing the vector representations of inputs, the model performs a question-answering task through a series of transformer layers. Each layer employs an attention mechanism, aiding in discerning relationships between vector representations of inputs.

• Attention Mechanism : This mechanism contributes to determining relationships between vector representations of inputs, enhancing the model's understanding of the contextual nuances.

• Prediction Layer : This layer predicts the answer to the question, consolidating the model's comprehension and providing a valuable output.

This innovative depiction underscores the intricate process of the DistilBert QA model, showcasing its proficiency in discerning relationships and delivering accurate responses to posed questions through the adept utilization of attention mechanisms and

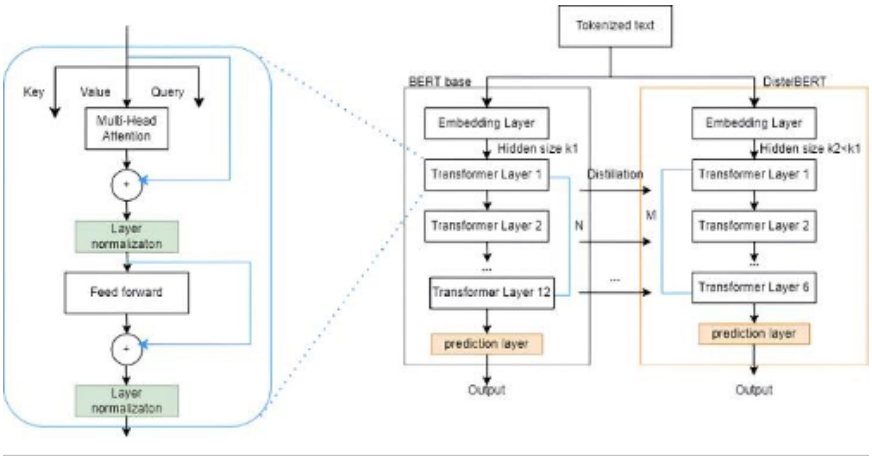

Figure 6 Example of how DistilBert works

transformer layers.

Finally, distinctions between the DistilBert QA model and the BERT base model can be outlined. Notably, the DistilBert model is smaller than the BERT base model. This implies that DistilBert has fewer parameters, resulting in a faster and more efficient operation. The DistilBert QA model proves to be a potent and efficient model for performing question-answering tasks. Its compact size, coupled with high performance, renders it suitable for a diverse range of applications.

RoBERTa: RoBERTa is a language model derived from BERT, building upon the masked language modeling (MLM) task at the core of BERT. However, RoBERTa exhibits significant differences from the original BERT model. Dynamic masking strategies are employed during the model's training, rendering the masking operations applied to input sequences more effective. This enhances the model's overall language understanding capability (Yasunaga *et al.* 2021).

A notable feature in RoBERTa's model architecture is the absence of the "Next Sentence Prediction" task. Instead, the model focuses on more specific tasks aimed at understanding relationships between sentences. This contributes to a better understanding of the context between sentences.

During training, RoBERTa is nourished with large datasets such as BooksCorpus, CC-News, OpenWebText, enriching the model's language learning abilities. Consequently, the model acquires more robust representation capabilities over general language knowledge.

In the process of model integration and fine-tuning, RoBERTa is adapted to be fine-tuned for a specific task. This customization enables the model to exhibit improved performance in the target task using pretrained language representations. The foundational architecture of RoBERTa, with its adaptability to specific tasks and overall language understanding capabilities, makes it an effective tool for various natural language processing tasks.

Figure 7 illustrates the performance of the RoBERTa QA model in an antonym test. In this test, the model is presented with both an original question and its antonym counterpart—a question that is opposite in meaning to the original. For instance, if the original question is 'What is the capital of Turkey?' the antonym question might be 'What is not the capital of Turkey?'

The horizontal axis in the Figure 7 represents the accuracy rate of the original question, while the vertical axis represents the accuracy rate of the antonym question. As evident from the graph, the RoBERTa QA model exhibits a lower accuracy rate in the antonym test compared to the original question. This indicates that the model encounters more challenges in comprehending antonym questions and providing accurate responses.

Several factors may contribute to this observation. Firstly, the model might have learned that antonym questions are inherently more difficult than the original questions. Secondly, the model could face difficulties in grasping the meaning of antonym questions. Lastly, the model may not have effectively learned the diverse strategies required to answer antonym questions.

These findings underscore the necessity for enhancing the resilience of the RoBERTa QA model specifically against antonym questions. This improvement could be achieved through the model gaining a better understanding of the meaning of antonym questions, developing different strategies, or undergoing specialized training for handling antonym questions. Addressing these aspects would contribute to the model's overall robustness in handling questions related to antonyms.

ALBERT: ALBERT, often referred to as "A Lite BERT," is specifically designed for natural language processing (NLP) tasks. It aims to lighten the BERT model to make it more scalable. While preserving the learning capabilities of BERT, ALBERT achieves the training of larger models with fewer parameters by factorizing the parameters in the embedding layer. This factorization allows the model to learn more efficiently. ALBERT is particularly developed to operate effectively and efficiently in resource-constrained environments. Additionally, it includes variants with different model sizes (small, base, large, xlarge), providing users the flexibility to choose the model size based on their needs (An *et al.* 2023).

In essence, the ALBERT model endeavors to deliver more effective performance in widely used natural language processing tasks by maintaining the advantages of BERT while presenting a lighter and more scalable architecture. ALBERT can be a suitable option, especially for applications that need to operate in resource-limited devices or environments.

In this study, ALBERT is trained with the Squad v2 dataset, and its performance on the QA model is elaborated upon. The introduction of ALBERT within the Model Architecture section under the Methodology aims to provide a comprehensive understanding of its role and capabilities within the scope of this research.

Flow diagram in Figure 8 illustrates the training process of the ALBERT QA model. The model undergoes a two-stage training process:

- Pre-training : During the pre-training stage, the model is

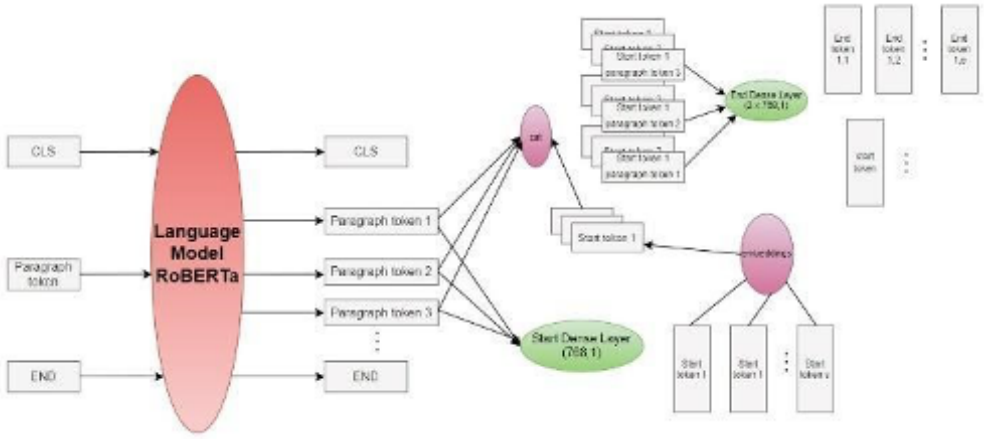

Figure 7 RoBERTa Model

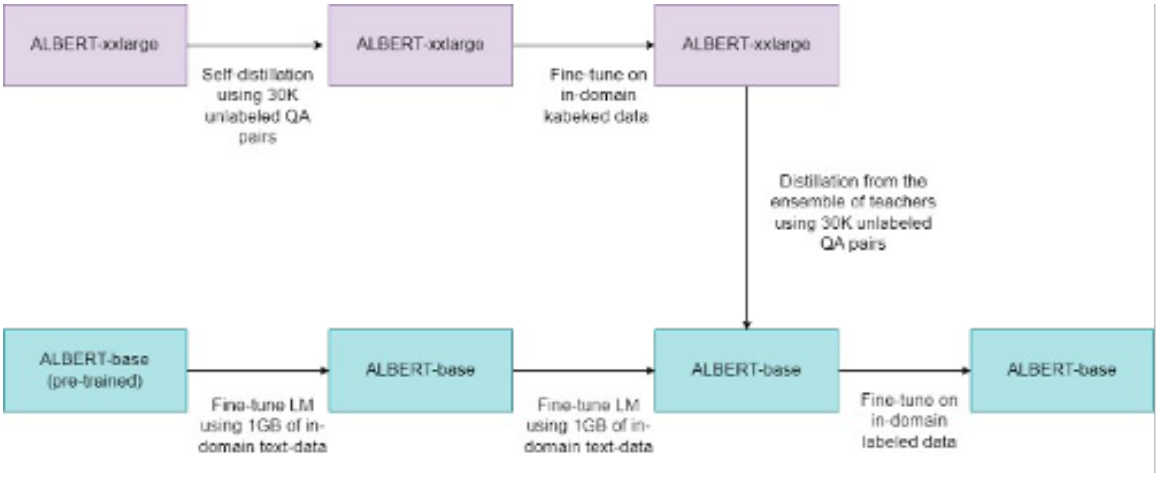

Figure 8 ALBERT QA Model Example

trained on a dataset comprising unlabeled text and question pairs. This dataset should encapsulate a representation of realworld questions and answers. By training on this dataset, the model learns to understand the relationships between text and questions (Tripathy *et al.* 2021).

• Fine-tuning : In the fine-tuning stage, the model is trained on a labeled dataset. This dataset includes question-answer pairs alongside accuracy labels. Training on this dataset allows the model to learn how to process text to correctly answer questions (Tripathy *et al.* 2022).

The pre-training stage enables the model to grasp its fundamental features. Here, the model learns to understand the relationships between text and questions, facilitating its comprehension to accurately answer questions.

In Figure 8, the pre-training stage is depicted as follows:

• ALBERT-xxlarge : The largest version of the model with 137B parameters.

• Self-distillation : The process of the model training on itself, employed to enhance its performance.

• 30K unlabeled QA pairs : The number of unlabeled question-answer pairs used for the pretraining of the model.

Self-distillation is a crucial process during which one version of the model is used to train another, contributing to the model's improved representation.

The subsequent fine-tuning stage allows the model to enhance its performance in a specific domain. In this stage, the model is trained on a labeled dataset that includes question-answer pairs and accuracy labels.

In Figure 8, the fine-tuning stage is depicted as follows:

• ALBERT-base : A smaller version of the model with 117B parameters.

• Fine-tune on in-domain labeled data : The process of the model training on itself, employed to enhance its performance.

• 30K unlabeled QA pairs : The process of training the model on a labeled dataset to ensure accurate answers to questions within a specific domain. The size of the labeled dataset used for the model's fine-tuning, significantly impacting its performance. A larger dataset contributes to the model's improved performance.

Training Process

BERT, ALBERT, and DistilBERT models were trained using the Squad v2 dataset. The training process involved fine-tuning pre-trained models to adapt their general language understanding capabilities to a specific QA task. Training data consisted of text paragraphs from the Squad v2 dataset along with various question-answer pairs directed towards these paragraphs. This process aimed to enhance the models' performance in specific topics or contexts.

Performance Metrics

The evaluation of BERT, ALBERT, and DistilBERT models was conducted using QA tasks on the Squad v2 dataset, and the assessment criteria were refined to encompass novel performance metrics. In addition to the traditional measures of accuracy, precision, recall, and F1 score, the evaluation now includes more specific metrics tailored to the nature of question-answering tasks.

The revised metrics comprise the following key elements:

• HasAns_exact : The count of responses that precisely provide an exact answer to the posed question.

• NoAns_exact : The count of responses that do not furnish an exact answer to the question.

• HasAns_partial : The count of responses that offer a partial answer to the question.

• NoAns_partial : The count of responses that fail to provide a partial answer to the question.

• HasAns_not_found : The count of responses that do not present any answer to the question.,

• Exact_match : The proportion of responses that provide an exact answer to the question.

Additionally, novel metrics are introduced to offer a more nuanced evaluation:

• F1_score : The proportion of responses that deliver both a correct and complete answer to the question.

• Best_f1_tresh : The threshold value at which the F1_score is maximized.

• Best_f1 : The highest value of the F1_score.

• Best_exact : The highest value of the Exact_match.

• Best_exact_tresh : The threshold value at which the Exact_match is maximized.

These refined metrics provide a comprehensive and nuanced assessment of the models' performance on the Squad v2 dataset, offering insights into their abilities to produce exact and partial answers, as well as highlighting their performance at different threshold values. This enhanced evaluation aims to better capture the distinctive features of each model and contribute to a more nuanced understanding of their effectiveness in addressing the challenges posed by the Squad v2 dataset.

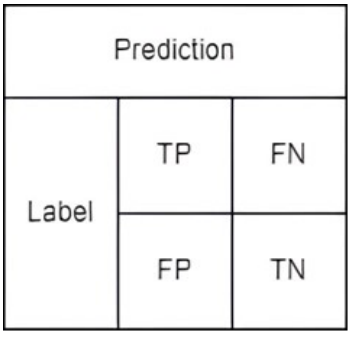

Figure 9 Confusion Matrix Explained

Accuracy Value: The accuracy value is a crucial performance metric that measures the percentage of correct predictions among the total predictions made by a model. It is mathematically expressed as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

• TP (True Positive) : The number of correct positive predictions.

• TN (True Negative) : The number of correct negative predictions.

• FP (False Positive) : The number of incorrect positive predictions.

• FN (False Negative) : The number of incorrect negative predictions.

Accuracy is employed to assess the overall performance of a model. However, it can be misleading in the presence of imbalanced class distributions. This metric is crucial in evaluating how much of the model's predictions are correct. Especially when dealing with imbalanced datasets, accuracy should be considered alongside other performance metrics. What is TP, TN, FP, FN is explained in Figure 9.

F1 Score: The F1 Score is a significant performance metric that strikes a balance between precision and recall, providing a comprehensive evaluation of a model's ability to make accurate positive predictions while minimizing false positives and false negatives. Mathematically, it is expressed as:

$$F1\,\text{Score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{2}$$

Here:

• Precision: Measures the accuracy of positive predictions, calculated as

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

• Recall: Gauges the model's ability to capture all positive instances, calculated as

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

RESULTS

During the evaluation of QA models on the SQuAD v2 dataset, each trained for 3 epochs, ALBERT emerged as the top performer, achieving an impressive 86.85% exact match and an 89.91% F1 score. Notably, ALBERT excelled in both answerable ('HasAns') and unanswerable ('NoAns') questions, demonstrating superior capabilities in providing accurate and comprehensive responses. BERT closely followed, demonstrating strong performance with a 65.96% exact match and a 70.12% F1 score. Its proficiency is particularly evident in answerable questions, where it achieved a remarkable 76.13% F1 score. RoBERTa secured robust results, yielding a 79.87% exact match and an 82.91% F1 score. Its balanced performance across answerable and unanswerable questions underscores its reliability in diverse QA scenarios. DistilBERT, while exhibiting competitive results with a 64.89% exact match and a 68.18% F1 score, falls slightly behind the other models. Nevertheless, it provides valuable insights, especially in scenarios where computational resources are constrained. In summary, the comprehensive assessment showcases ALBERT as the standout performer, followed closely by BERT and RoBERTa, while DistilBERT, although slightly trailing, remains a viable option for resource-efficient applications. You can see the values we have discussed and explained here in Table 1 comparatively.

Table 1 Results of All Models

Model	BERT-medium	DistilBERT	RoBERTa	ALBERT
F1	70.1163	68.1776	82.9125	89.9148
exact	65.9564	64.8898	79.8703	86.8525
total	11873	6078	11873	11873
HasAns_exact	67.7969	69.7595	77.9352	84.4467
HasAns_f1	76.1287	76.6267	84.0284	90.5801
HasAns_total	5928	2910	5928	5928
NoAns_exact	64.1211	60.4167	81.7998	89.2515
NoAns_f1	64.1211	60.4167	81.7998	89.2515
NoAns_total	5945	3168	5945	5945
Best_exact	65.9648	64.8898	79.8703	87.4168
Best_exact_thresh	0	0	0.95	-3.0903
Best_f1	70.1247	68.1776	82.9125	90.3287
Best_f1_thresh	0	0	0.95	-3.0903

CONCLUSION

The performances of DistilBERT, BERT, ALBERT, and RoBERTa models on the SQuAD v2 dataset were evaluated after three epochs of training in this study. The 'exact' and 'f1' scores were analyzed to compare the models. DistilBERT was found to lag slightly behind other models, exhibiting 'exact' and 'f1' scores of 64.89% and 68.18%, respectively. Despite this, it emerged as a valuable option in scenarios where computational resources are limited. Higher 'exact' and 'f1' scores were observed for BERT, standing out at 65.96% and 70.12%, respectively. Particularly impressive was its performance in answerable questions, contributing to an overall increased F1 score. ALBERT drew attention with stellar performance, demonstrating 'exact' and 'f1' scores of 86.85% and 89.91%, respectively. It showcased significant superiority over other models in both answerable and unanswerable questions. RoBERTa exhibited a balanced performance, achieving 'exact' and 'f1' scores of 79.87% and 82.91%, respectively, with consistent results for both answerable and unanswerable questions. The limited number of epochs may have contributed to DistilBERT generally showing lower performance, while the ALBERT model, showcasing rapid learning ability with a small number of epochs, surpassed other models.

BERT's transformer architecture allows it to achieve state-of-the-art results by pre-training on a large corpus and fine-tuning on specific tasks. The model employs a bidirectional approach, considering the context from both directions, which sets it apart from traditional models like LSTM and GRU. The SQuAD (Stanford Question Answering Dataset) has been a benchmark for evaluating question-answering models. The release of SQuAD v2 introduced unanswerable questions, adding a new challenge for models to determine when no answer is possible.

However, other factors should also be considered when making rankings. Model performances should be evaluated, particularly based on specific usage scenarios. For instance, while DistilBERT might be a valuable option in scenarios where computational resources are limited, BERT's impressive performance in answerable questions might make it a preferable choice in such scenarios. This study highlights the importance of hyperparameter optimization in fine-tuning pretrained models. The results suggest that fine-tuning BERT with appropriate hyperparameters can lead to significant performance gains. This insight contributes to the broader understanding of how pre-trained models can be effectively adapted to specific tasks.

In conclusion, an important guide for the use of models in various QA scenarios is provided by this evaluation, considering their performances. The advantages and disadvantages of each model should be assessed based on specific use cases.

In this study, the performances of four different BERT models - RoBERTa, ALBERT, DistilBERT, and BERT - in the field of Question Answering (QA) were compared after being trained for 3 epochs. Evaluation criteria included "exact" and "f1" scores. The obtained results indicate significant differences among the models. Initially, it was deemed necessary to rank the models based on the evaluation scores.

The ranking, according to the comparison values, is as follows: ALBERT - RoBERTa - BERT - DistilBERT. ALBERT, especially by demonstrating faster learning capability with fewer epochs, exhibits superior performance compared to other models. This advantage of ALBERT stems from the limited duration of the training. If the epoch numbers of other models are increased, these models might approach or surpass the accuracy rate achieved by ALBERT.

This emphasizes the importance of strategic decisions related to the training duration of models.

However, other factors should also be considered when making rankings. Model performances should be evaluated, particularly based on specific usage scenarios. For instance, while DistilBERT might be a valuable option in scenarios where computational resources are limited, BERT's impressive performance in answerable questions might make it a preferable choice in such scenarios. In conclusion, this evaluation provides a valuable resource for understanding the strengths and weaknesses of each model. It guides researchers in determining which model might be more effective in specific QA scenarios. Future studies could delve more deeply into how these models can be further enhanced with additional features and parameter adjustments and how they perform on different datasets.

Exploring ensemble approaches that combine BERT with other models could also yield interesting results. The implications of these findings are substantial for the development of more sophisticated AI-driven question-answering systems. By improving the model's ability to handle unanswerable questions, we enhance the reliability and user trust in AI systems. This has practical applications in customer service, virtual assistants, and information retrieval systems, where accurately identifying unanswerable questions can prevent misinformation and improve user experience.

Availability of data and material

Not applicable.

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The authors have no relevant financial or non-financial interests to disclose.

LITERATURE CITED

An, Q., B. Pan, Z. Liu, S. Du, and Y. Cui, 2023 Chinese named entity recognition in football based on albert-bilstm model. Applied Sciences **13**: 10814.

Benedetto, L., 2023 A quantitative study of nlp approaches to question difficulty estimation. In *International Conference on Artificial Intelligence in Education*, pp. 428–434, Springer Nature Switzerland.

Caballero, M., 2021 A brief survey of question answering systems. International Journal of Artificial Intelligence & Applications (IJAIA) **12**.

David, J., 2020 Comparing the representation learning of autoencoding transformer models in ad hoc information retrieval .

Devlin, J. *et al.*, 2018 Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 .

Fu, X., J. Du, H. T. Zheng, J. Li, C. Hou, *et al.*, 2023 Ss-bert: A semantic information selecting approach for open-domain question answering. Electronics **12**: 1692.

Ghanem, R., H. Erbay, and K. Bakour, 2023 Contents-based spam detection on social networks using roberta embedding and stacked blstm. SN Computer Science **4**: 380.

Gillioz, A., J. Casas, E. Mugellini, and O. Abou Khaled, 2020 Overview of the transformer-based models for nlp tasks. In *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, pp. 179–183, IEEE.

Greco, C. M., A. Tagarelli, and E. Zumpano, 2022 A comparison of transformer-based language models on nlp benchmarks. In *International Conference on Applications of Natural Language to Information Systems*, pp. 490–501, Springer International Publishing.

Kasai, J. *et al.*, 2024 Realtime qa: What's the answer right now? Advances in Neural Information Processing Systems **36**.

Kumar, A., T. Ranjan, and S. Raghav, 2023 Building conversational question answer machine and comparison of bert and its different variants. In *2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pp. 240–245, IEEE.

Kumari, V., S. Keshari, Y. Sharma, and L. Goel, 2022a Context-based question answering system with suggested questions. In *2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 368–373, IEEE.

Kumari, V., Y. Sharma, and L. Goel, 2022b A comparative analysis of transformer-based models for document visual question answering. In *International Conference on Computational Intelligence and Data Engineering*, pp. 231–242, Springer Nature Singapore.

Lan, Z. *et al.*, 2019 Albert: A lite bert for self-supervised learning of language representations.

Liu, Y. *et al.*, 2019 Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 .

MacRae, C., 2022 *NOLEdge: Creating an Intelligent Search Tool for the Florida State University Computer Science Department Using Fine-Tuned Transformers and Data Augmentation*.

Malla, S. and P. J. A. Alphonse, 2021 Covid-19 outbreak: An ensemble pre-trained deep learning model for detecting informative tweets. Applied Soft Computing **107**: 107495.

Nassiri, K. and M. Akhloufi, 2023 Transformer models used for text-based question answering systems. Applied Intelligence **53**: 10602–10635.

Pereira, J., R. Fidalgo, R. Lotufo, and R. Nogueira, 2023 Visconde: Multi-document qa with gpt-3 and neural reranking. In *European Conference on Information Retrieval*, pp. 534–543, Springer Nature Switzerland.

Pirozelli, P., A. A. Brandão, S. M. Peres, and F. G. Cozman, 2022 To answer or not to answer? filtering questions for qa systems. In *Brazilian Conference on Intelligent Systems*, pp. 464–478, Springer International Publishing.

Rawat, A. and S. S. Samant, 2022 Comparative analysis of transformer based models for question answering. In *2022 2nd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, pp. 1–6, IEEE.

Sabharwal, N. and A. Agrawal, 2021 *Hands-on Question Answering Systems with BERT: Applications in Neural Networks and Natural Language Processing*.

Schütz, M., A. Schindler, M. Siegel, and K. Nazemi, 2021 Automatic fake news detection with pretrained transformer models. In *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10-15, 2021, Proceedings, Part VII*, pp. 627–641, Springer International Publishing.

Sidorov, G., F. Balouchzahi, S. Butt, and A. Gelbukh, 2023 Regret and hope on transformers: An analysis of transformers on regret and hope speech detection datasets. Applied Sciences **13**: 3983.

Srivastava, V., S. Pilli, S. Bhat, N. Pedanekar, and S. Karande, 2021 What berts and gpts know about your brand? probing contextual language models for affect associations. In *Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pp. 119–128.

Sundelin, C., 2023 Comparing different transformer models' performance for identifying toxic language online .

Tahsin Mayeesha, T., A. Md Sarwar, and R. M. Rahman, 2021 Deep learning based question answering system in bengali. Journal of Information and Telecommunication **5**: 145–178.

Tripathy, J. K., S. S. Chakkaravarthy, S. C. Satapathy, M. Sahoo, and V. Vaidehi, 2022 Albert-based fine-tuning model for cyberbullying analysis. Multimedia Systems **28**: 1941–1949.

Tripathy, J. K., S. C. Sethuraman, M. V. Cruz, A. Namburu, P. Mangalraj, *et al.*, 2021 Comprehensive analysis of embeddings and pre-training in nlp. Computer Science Review **42**: 100433.

Wang, N., R. R. Issa, and C. J. Anumba, 2022 Nlp-based query-answering system for information extraction from building information models. Journal of computing in civil engineering **36**: 04022004.

Yang, Z. *et al.*, 2019 Xlnet: Generalized autoregressive pretraining for language understanding. Advances in neural information processing systems **32**.

Yasunaga, M., H. Ren, A. Bosselut, P. Liang, and J. Leskovec, 2021 Qa-gnn: Reasoning with language models and knowledge graphs for question answering. arXiv preprint arXiv:2104.06378 .

Yu, M. and A. Sun, 2023 Dataset versus reality: Understanding model performance from the perspective of information need. Journal of the Association for Information Science and Technology **74**: 1293–1306.

How to cite this article: Ozkurt, C. Comparative Analysis of State-of-the-Art Q&A Models: BERT, RoBERTa, DistilBERT, and ALBERT on SQuAD v2 Dataset. *Chaos and Fractals*, 1(1), 19-30, 2024.