Journal of Universal Computer Science, vol. 32, no. 11 (2026), 1650-1670
submitted: 24/10/2025, accepted: 5/2/2026, appeared: 28/10/2026 CC BY 4.0

Comparative Analysis of Transformer-Based Models for Question Answering on SQuAD v2.0

Oğuzhan Kırlar
(Department of Computer Engineering, Faculty of Engineering, Kocaeli University, Kocaeli, Turkey
https://orcid.org/0009-0006-2023-1457, oguzhan.kirlar@gmail.com)

Mehmet Ali Altuncu
(Department of Computer Engineering, Faculty of Engineering, Kocaeli University, Kocaeli, Turkey
https://orcid.org/0000-0002-2948-3937, mehmetali.altuncu@kocaeli.edu.tr)

Abstract: Transformer-based architectures have revolutionized Question Answering (QA), yet navigating the trade-off between architectural complexity, computational efficiency, and robust generalization remains a critical challenge. This study presents a rigorous, side-by-side evaluation of four distinct Transformer models—ALBERT, RoBERTa, DeBERTa, and XLNet—on the SQuAD v2.0 benchmark, which demands both precise span extraction and calibration to abstain from unanswerable queries. By employing identical preprocessing pipelines and hyperparameter schedules, we isolate the impact of specific architectural innovations, including dynamic masking, disentangled attention, and permutation language modeling. Experimental results establish RoBERTa as the performance leader (85.90% EM, 88.69% F1), validating the efficacy of robust optimization strategies. Notably, DeBERTa followed closely (85.41% EM, 88.54% F1) but achieved superior performance on unanswerable questions (88.24%), suggesting that its disentangled attention mechanism offers enhanced verification capabilities against distractor entities. XLNet demonstrated competitive potential (78.30% EM) via its autoregressive formulation, while ALBERT confirmed its status as the efficiency benchmark, delivering respectable accuracy (80.70% EM) with a fraction of the parameter footprint. Qualitative error analysis reveals that while lighter models are prone to "hallucinating" answers, advanced architectures primarily fail due to subtle reasoning errors rather than entity mismatch. These findings provide a definitive decision framework for practitioners: prioritizing RoBERTa and DeBERTa for maximum reliability in complex domains, and leveraging ALBERT for scalable, resource-constrained deployments.

Keywords: Question Answering, Transformer Models, ALBERT, RoBERTa, DeBERTa, XLNet
Categories: I.4.7, I.5, I.6.4, I.2.6
DOI: 10.3897/jucs.175976

1 Introduction

In recent years, the development of models capable of understanding human language and providing accurate answers has become a central focus in both academic research and industrial applications of Natural Language Processing (NLP). Question Answering (QA) systems have emerged as crucial tools for enabling fast and reliable access to information. Unlike traditional information retrieval systems, which simply return a set of documents, modern QA systems are designed to pinpoint precise answers

within large text corpora. [Chen et al., 2017] This capability makes them increasingly valuable in a wide range of domains, from search engines and digital assistants to customer support systems, healthcare information retrieval, and education technology. As the demand for human–machine interaction grows, QA has become one of the most actively studied tasks in NLP [Young et al., 2018].

Among the available benchmarks, the Stanford Question Answering Dataset (SQuAD) has played a significant role in shaping the development of QA models. The original release, SQuAD v1.1, provided over 100,000 question–answer pairs based on passages from Wikipedia. However, it contained only answerable questions, making it somewhat limited for real-world applications where not all questions have a valid answer in the given context. To address this limitation, SQuAD v2.0 introduced over 50,000 unanswerable questions in addition to the original set. This extension makes the benchmark substantially more challenging, as models must not only extract correct spans when answers exist but also correctly abstain from answering when they do not. The ability to recognize when no answer is available is critical for practical deployment, as users expect systems not to provide misleading or fabricated outputs. Consequently, SQuAD v2.0 has become the de facto standard for evaluating the robustness of extractive QA models. [Rajpurkar et al., 2018]

Within this evolving landscape, Transformer-based models have established themselves as the backbone of modern NLP. Since the introduction of Transformer architecture by [Vaswani et al., 2017], models such as BERT, GPT, and their successors have achieved state-of-the-art results across diverse tasks. [Devlin et al., 2019] The self-attention mechanism at the heart of Transformers allows these models to capture long-range dependencies in text, offering significant improvements over earlier recurrent or convolutional approaches. Building on BERT's success, researchers have proposed a wide range of extensions aimed at improving either performance or efficiency, leading to a diverse ecosystem of models tailored to specific trade-offs.

Four such models that have attracted considerable attention in the context of QA and represent distinct design philosophies are ALBERT, RoBERTa, DeBERTa, and XLNet. While they all build upon the Transformer foundation, they diverge in their architectural priorities:

ALBERT (A Lite BERT) introduces parameter-reduction techniques, most notably factorized embedding parameterization and cross-layer parameter sharing—that dramatically reduce the total number of trainable parameters without severely compromising representational capacity. This makes ALBERT a highly parameter-efficient model, appealing for scenarios where memory and computational constraints are significant. [Lan et al., 2020]

RoBERTa (Robustly Optimized BERT Pretraining Approach) focuses on robust optimization. It eliminates the Next Sentence Prediction (NSP) objective, leverages much larger pretraining corpora, increases batch sizes, and employs dynamic masking. These adjustments yield substantial accuracy improvements, positioning RoBERTa as a strong baseline for high-accuracy applications. [Liu et al., 2019]

DeBERTa (Decoding-enhanced BERT with disentangled attention) improves upon RoBERTa by introducing a disentangled attention mechanism, where content and relative position are encoded separately. Coupled with an enhanced mask decoder, DeBERTa captures subtler syntactic dependencies, aiming to surpass the performance of standard BERT variants. [He et al., 2021]

XLNet diverges from the auto-encoding framework of the others by employing Permutation Language Modeling (PLM). By maximizing the expected likelihood over all permutations of the factorization order, XLNet captures bidirectional contexts while retaining the autoregressive advantages of Transformer-XL, theoretically offering better handling of long-range dependencies. [Yang et al., 2019]

Although these models have been extensively studied in isolation, comparative analyses that directly contrast this diverse set of architectures—spanning parameter-efficient, robustly optimized, architecturally enhanced, and permutation-based approaches—under controlled conditions remain limited. Existing literature often evaluates them independently or in broader model comparisons with varying hyperparameters, but few studies have systematically analyzed their relative strengths and weaknesses specifically in challenging QA tasks like SQuAD v2.0. This gap is particularly important because practical deployment frequently requires balancing accuracy against efficiency. [Özkurt, 2024; Harikrishnan et al., 2025] Understanding how these models behave under identical training and evaluation setups, and where one may be favored over the other, provides valuable guidance for both researchers and practitioners.

This study addresses these gaps by conducting a detailed comparative evaluation of ALBERT, RoBERTa, DeBERTa, and XLNet on the SQuAD v2.0 dataset. Specifically, performance is examined in terms of Exact Match (EM) and F1 scores, alongside analyses of training efficiency, model complexity, and resource consumption. In addition to quantitative outcomes, qualitative insights into error patterns are provided, and the implications for real-world deployment are discussed. By combining a rigorous experimental design with both accuracy and efficiency considerations, this work contributes to the ongoing development of high-performance and practical QA systems. Our findings highlight not only the individual capabilities of each model but also the strategic choices that researchers and practitioners must make when selecting models for different applications. In doing so, the study bridges the gap between academic benchmarking and real-world deployment, offering actionable insights for the design of next-generation QA systems.

The main contributions of this study can be summarized as follows:

i. The establishment of a unified comparative framework that evaluates four distinct Transformer architectures—ALBERT, RoBERTa, DeBERTa, and XLNet—under identical experimental conditions, thereby isolating the impact of architectural differences on question answering performance.
ii. In addition to quantitative benchmarking, the study systematically analyzes trade-offs between accuracy and efficiency, and between computational cost and performance, across Transformer architectures, providing evidence-based criteria for model selection in resource-constrained and high-performance environments.
iii. Furthermore, a detailed qualitative error analysis is conducted to identify failure modes in both answerable and unanswerable scenarios.
iv. Finally, the calibration of abstention strategies and robustness against distractor entities are assessed to examine how architectural enhancements contribute to improved reliability in real-world deployment.

2 Literature Review

The introduction of the Transformer architecture [Vaswani et al., 2017] revolutionized NLP by replacing recurrent [Hochreiter and Schmidhuber, 1997] and convolutional structures with attention-based mechanisms, enabling efficient parallelization and long-range dependency modeling. Among Transformer-based models, BERT [Devlin et al., 2019] became the first widely adopted model for question answering, achieving state-of-the-art results on SQuAD and inspiring numerous variants tailored toward efficiency and accuracy.

ALBERT (A Lite BERT) was proposed by [Lan et al., 2020] as a parameter-reduction variant of BERT. Through factorized embedding parameterization and cross-layer parameter sharing, ALBERT reduces model size substantially while retaining competitive accuracy. Large-scale experiments demonstrated that ALBERT-xxlarge could achieve near state-of-the-art results, with an EM close to 90% and F1 above 92% on SQuAD v2.0 [Lan et al., 2020]. Its efficiency makes it particularly suitable for large-scale or resource-constrained applications. Empirical evaluations such as those of [Özkurt, 2024] confirmed ALBERT's capability, reporting an EM of 86.85% and F1 of 89.91%.

RoBERTa (Robustly Optimized BERT Pretraining Approach) extended BERT by refining its pretraining procedure [Liu et al., 2019]. It removed the Next Sentence Prediction objective, increased training data, employed dynamic masking, and used larger batch sizes. These optimizations resulted in consistent improvements across multiple tasks, including SQuAD v2.0. RoBERTa-large has been ranked among the strongest QA models, with EM scores above 90% and F1 exceeding 94% in large-scale evaluations [He et al., 2021]. More recent comparative studies, reported that RoBERTa maintains superiority over lighter models like DistilBERT [Sanh et al., 2019], with an EM of 80.01% and F1 of 82.47%, underscoring its robustness in QA tasks.

Beyond ALBERT and RoBERTa, several alternative architectures have been proposed. XLNet [Yang et al., 2019] combined permutation language modeling with Transformer encoders, achieving an EM of 87.9% and F1 of 90.6% on SQuAD v2.0. XLNet demonstrated strong performance on the SQuAD benchmark through advanced training and fine-tuning strategies. DeBERTa, another variant focusing on disentangled attention mechanisms, reported an EM of 86.2% and F1 of 83.1% for its base version, and an EM of 89.0% and F1 of 91.5% for the DeBERTa-v3-large variant [He et al., 2021; He et al., 2023], which utilizes Electra-style pre-training [Clark et al., 2020]. MiniLM, introduced by [Wang et al., 2020], offered a distilled architecture with an EM between 86–88% on SQuAD v2.0, representing a practical trade-off between compactness and accuracy. More recently, [Harikrishnan et al., 2025] evaluated a BERT-based QA pipeline on SQuAD, reporting an F1 of 79.37%, further highlighting how pipeline design influences performance. Collectively, these studies underscore both the accuracy–efficiency spectrum and the continuous innovation in adapting Transformer architectures to QA.

Although ALBERT, RoBERTa, DeBERTa, and XLNet each perform strongly on QA tasks, direct comparative analyses that systematically contrast this diverse set of architectures under identical conditions remain limited. While existing literature often evaluates these models in isolation or highlights specific trade-offs—such as ALBERT's efficiency versus RoBERTa's accuracy—few studies have

comprehensively analyzed how autoregressive mechanisms (XLNet) and disentangled attention (DeBERTa) compare against standard autoencoding models within a unified experimental framework. Addressing these gaps requires structured evaluations that not only quantify performance through EM and F1 but also consider computational overhead, architectural nuances, and error patterns to inform real-world deployment decisions.

3 Methodology

In this study, the experiments were carried out using the SQuAD v2.0 dataset, which contains both answerable and unanswerable questions and serves as a widely adopted benchmark for extractive question answering. The dataset is explained in detail in Section 4, while the following subsections present the architectural and methodological characteristics of ALBERT, RoBERTa, DeBERTa, and XLNet, together with the enhancements applied during their pretraining and architectural design phases.

3.1 Albert

ALBERT [Lan et al., 2020] was introduced to address the problem of parameter inefficiency in large-scale language encoders while preserving their representational capacity. Instead of assigning unique parameters to each Transformer layer, ALBERT applies cross-layer parameter sharing, in which the same set of weights is reused across multiple layers. This approach drastically reduces the number of trainable parameters, lowers memory requirements, and provides a form of implicit regularization that can enhance stability during fine-tuning. Another key innovation is factorized embedding parameterization, where the large word embedding matrix is decoupled from the hidden layer dimension. Embeddings are first projected into a lower-dimensional space before being expanded back to the hidden size, enabling efficient learning without loss of semantic richness. Additionally, ALBERT replaces BERT's Next Sentence Prediction (NSP) objective with a Sentence Order Prediction (SOP) task. By requiring the model to determine whether two consecutive sentences are in the correct order or reversed, SOP captures discourse-level coherence more effectively than NSP. Together, these modifications allow ALBERT to achieve high accuracy with significantly fewer parameters.

3.2 RoBERTa

RoBERTa (Robustly Optimized BERT Pretraining Approach), introduced by [Liu et al., 2019], replicates the BERT architecture but implements critical modifications to the pretraining procedure to enhance performance. Unlike BERT's static masking, RoBERTa employs dynamic masking, generating a unique masking pattern for every input sequence during training epochs, which increases data variability. Furthermore, it eliminates the Next Sentence Prediction (NSP) objective, having found it to be redundant for downstream performance. The model is trained on a substantially larger corpus (approx. 160GB) with significantly larger batch sizes and for longer durations. These optimizations result in more robust contextual representations, allowing RoBERTa to consistently outperform BERT on benchmarks such as SQuAD.

3.3 DeBERTa

DeBERTa (Decoding-enhanced BERT with disentangled attention), proposed by [He et al., 2021], improves upon the BERT and RoBERTa architectures through two novel techniques: a disentangled attention mechanism and an enhanced mask decoder. Unlike previous models that represent each token with a single vector combining content and position, DeBERTa encodes content and position separately using two distinct vectors. The attention mechanism computes scores based on disentangled matrices for content and relative positions, allowing the model to capture the dependency between token pairs more effectively. Additionally, DeBERTa incorporates an enhanced mask decoder that injects absolute position embeddings in the decoding layer (rather than just the input), which is crucial for predicting masked tokens based on their syntactic roles. These architectural innovations enable DeBERTa to achieve superior performance on natural language understanding tasks and question answering benchmarks.

3.4 XLNet

XLNet [Yang et al., 2019], a generalized autoregressive pretraining method, was introduced to overcome the limitations of BERT's masked language modeling (MLM) approach. Unlike BERT, which corrupts the input with [MASK] tokens that are absent during fine-tuning (creating a pretrain-finetune discrepancy), XLNet employs Permutation Language Modeling (PLM). This technique allows the model to learn bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order, rather than masking fixed tokens. Furthermore, XLNet integrates the autoregressive advantages of Transformer-XL, including the recurrence mechanism and relative positional encoding, enabling it to capture longer-term dependencies more effectively than standard autoencoding models like BERT and RoBERTa.

3.5 Comparative characteristics of models

The comparison of ALBERT, RoBERTa, DeBERTa, and XLNet reveals distinct trade-offs between architectural complexity, computational efficiency, and downstream performance. ALBERT stands out for its extreme parameter efficiency (~12M parameters), achieved through cross-layer sharing and embedding factorization, making it the optimal choice for memory-constrained environments. RoBERTa (~125M parameters) prioritizes robust optimization via dynamic masking and large-scale training corpora, establishing a strong baseline for accuracy. DeBERTa improves upon this foundation by introducing disentangled attention mechanisms and an enhanced mask decoder, aiming to surpass RoBERTa's performance by capturing subtler syntactic dependencies without a prohibitive increase in model size. Finally, XLNet diverges from the autoencoding framework of the others by employing Permutation Language Modeling (PLM), offering advantages in capturing long-range dependencies but often requiring higher computational overhead. Consequently, the selection among these models depends on deployment priorities: ALBERT for speed and scalability, whereas RoBERTa, DeBERTa, and XLNet are preferable when maximizing accuracy is the paramount objective.

3.6 Performance metrics

When evaluating model performance in extractive question answering (SQuAD-style contextual question answering), the chosen metrics must not only reflect the correctness of the predicted answer but also capture the degree of overlap between the prediction and the reference text. This aspect becomes particularly critical in datasets such as SQuAD v2.0, which include both answerable and unanswerable questions. The careful selection of evaluation metrics is therefore essential for assessing the applicability of models in real-world scenarios. In this study, two widely recognized metrics—Exact Match (EM) and F1 score—were employed [Rajpurkar et al., 2018].

The Exact Match metric evaluates whether the model's predicted answer exactly matches one of the ground-truth answers in the dataset, after normalization procedures such as lowercasing, punctuation removal, and whitespace handling. If an exact match is achieved, the EM score for that question is 1 (100%); otherwise, it is 0. EM is thus a strict metric, offering a high-precision measure of success that is particularly relevant in critical information systems. However, its lack of flexibility may penalize semantically correct answers that differ slightly in surface form. For example, if the ground-truth answer is "Barack Obama" and the model predicts only "Obama", the EM score is 0, despite the semantic proximity.

The F1 score provides a more flexible measure by evaluating the degree of token overlap between the predicted and ground-truth answers. Precision is computed as the proportion of predicted tokens that are present in the reference answer, while recall measures the proportion of reference tokens captured by the prediction. The harmonic mean of these two measures yields the F1 score, defined as:

$$\mathrm{f1} - \mathrm{Score} = (2xRecall * Precision)/(Recall + Precision) \quad (1)$$

By rewarding partial correctness, the F1 score accounts for semantically similar responses that may not exactly match the reference in form. This property is especially valuable in SQuAD v2.0, where multiple valid ground-truth answers may exist for a single question.

Within the SQuAD v2.0 evaluation framework, if no answer exists in the passage and the model correctly abstains from producing one, this outcome is counted as fully correct under both EM and F1. Conversely, producing an answer for an unanswerable question is considered a major error. Therefore, the decision of whether to return an answer is itself a crucial determinant of performance. Taken together, EM and F1 provide a complementary evaluation framework: EM captures exact correctness, while F1 measures contextual overlap and partial validity. For this reason, the performance of both individual models and ensemble strategies in this study was assessed using these two metrics.

4 Dataset

The Stanford Question Answering Dataset (SQuAD) is a high-quality, human-annotated benchmark created to evaluate contextual machine reading comprehension and extractive question answering performance within the field of natural language processing (NLP). The first version, SQuAD v1.1, was released in 2016 and consists of paragraphs from Wikipedia articles paired with crowd-sourced questions, each of

which has an answer span explicitly present within the corresponding passage [Rajpurkar et al., 2016]. While SQuAD v1.1 enabled substantial progress in question answering research, its structure lacked coverage of an important real-world phenomenon: the presence of questions that cannot be answered from the given text. To address this limitation, SQuAD v2.0 was introduced in 2018. This version retains all the original question–answer pairs from v1.1 but additionally includes unanswerable questions that are relevant to the passage yet have no valid answer span [Rajpurkar et al., 2018]. This design requires models not only to identify correct answer spans when they exist but also to abstain from answering when no valid response is supported by the text.

SQuAD v2.0 contains more than 150,000 questions in total, of which approximately 50,000 are explicitly labeled as unanswerable. The dataset is derived from 442 Wikipedia articles and about 19,000 passages, each passage averaging 122 words in length. For passages with answerable questions, the average answer length is 3.2 words, with a maximum observed length of 30 words. On average, each passage is associated with 5–6 questions, and answerable questions include approximately 1.8 distinct ground-truth variations, reflecting the natural linguistic diversity of possible responses. This structural diversity enables evaluation not only of information extraction capabilities but also of contextual understanding, recognition of multiple valid answers, and detection of unanswerable cases.

The dataset is divided into training, development, and test subsets, facilitating comparative experimentation. The training split contains approximately 130,000 questions, while the development set includes over 10,000 questions. Data instances are stored in JSON format, where each record consists of a title, one or more passages, and associated question–answer pairs. For answerable questions, the starting index of the answer span (answer_start) and its textual content are explicitly annotated. For unanswerable questions, the field "is_impossible": true is used to indicate the absence of an answer. Each question may be aligned with one or more correct answers, which must appear verbatim in the passage. SQuAD v2.0 is widely regarded as a reliable dataset in terms of annotation quality. Questions and answers were generated and validated by Amazon Mechanical Turk workers with careful attention to natural language flow, ensuring the dataset challenges models not only in span extraction but also in contextual awareness and safe response strategies.

Today, SQuAD v2.0 has become one of the most widely adopted benchmarks for evaluating transformer-based language models in question answering tasks. Models fine-tuned on this dataset are frequently used as reference points in both academic literature and industrial applications. Importantly, the presence of unanswerable questions enables assessment of advanced capabilities such as confidence calibration and abstention. As [Rajpurkar et al., 2018, p. 2] note, “SQuAD v2.0 tests a system’s ability not only to find information but also to know when no information is available.” In this sense, the dataset provides a more challenging and realistic benchmark that captures multiple dimensions of performance in modern extractive question answering systems.

5 Experimental Results

The experiments conducted on the SQuAD v2.0 dataset reveal distinct performance tiers among the evaluated architectures. RoBERTa achieved the highest accuracy with an EM score of 85.90% and an F1 score of 88.69%. DeBERTa followed closely, recording 85.41% EM and 88.54% F1, indicating that its disentangled attention mechanism offers comparable robustness to RoBERTa's dynamic masking strategy. XLNet demonstrated competitive capability with 78.30% EM and 81.24% F1. Although it trailed the autoencoding models in this specific setting, its permutation-based approach remains theoretically powerful for longer contexts. In contrast, ALBERT yielded 80.70% EM and 83.78% F1. While trailing the larger architectures in absolute accuracy, ALBERT confirmed its role as the most parameter-efficient model. Collectively, these results highlight a clear trade-off: RoBERTa and DeBERTa are the optimal choices for applications requiring maximum accuracy, whereas ALBERT remains the superior alternative for resource-constrained environments where memory footprint is the primary concern.

5.1 Preprocessing steps

To adapt the dataset for training, a series of preprocessing steps were applied to ensure compatibility with each model architecture. Both the training and validation subsets of SQuAD v2.0 were processed.

- Whitespace Normalization: Since tokenizer sensitivity varies across models, both questions and contexts were stripped of redundant whitespace to prevent token misalignment.
- Tokenization: Each question–context pair was tokenized using the model-specific tokenizer. RoBERTa utilizes a byte-level BPE tokenizer, whereas ALBERT, DeBERTa, and XLNet employ SentencePiece-based tokenizers.
- Truncation and Stride: Truncation was applied only to the context (with a stride of 128) to prevent answer spans from being truncated in long passages. Overlapping windows were generated to ensure full coverage of the input text.
- Offset Mapping: Token-to-character alignment was retained through offset mapping, which allows answer spans to be reconstructed precisely during training and evaluation.
- Answer Localization: For answerable questions, the character-level start and end indices were mapped to token positions. For unanswerable questions, both indices were set to zero (pointing to the [CLS] token or equivalent).
- Model-Specific Adjustments:
    - Segment IDs: For ALBERT, XLNet, and DeBERTa, token_type_ids were included to explicitly distinguish between question and context segments. In contrast, RoBERTa does not utilize segment embeddings; it relies solely on input_ids and special separator tokens (</s>) to discern segment boundaries.
    - Padding: Standard right-padding was applied for ALBERT, RoBERTa, and DeBERTa. For XLNet, the preprocessing pipeline

was adjusted to handle its specific padding requirements where necessary.

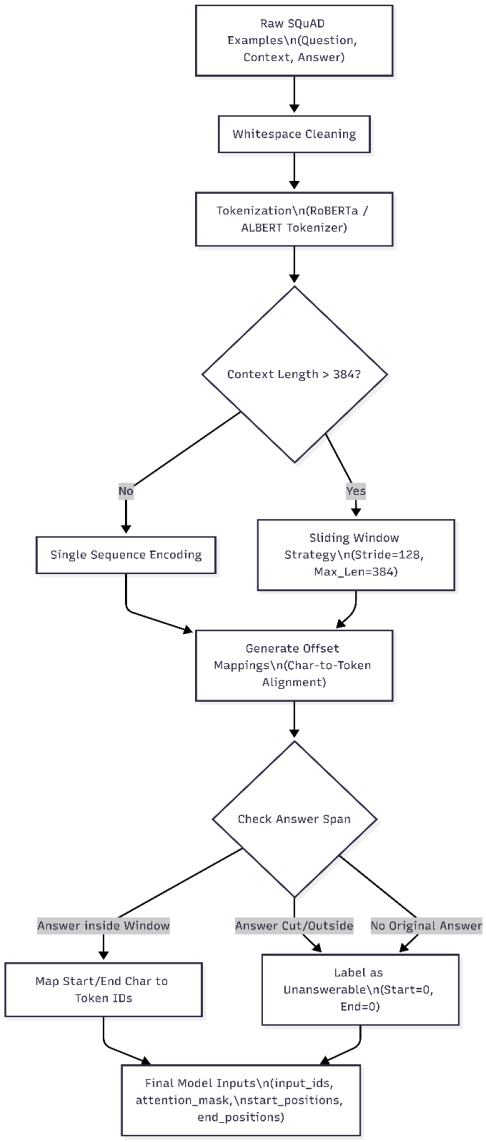

Figure 1: Schematic representation of the preprocessing pipeline

Following preprocessing, the tokenized data were converted into PyTorch datasets and loaded into DataLoader objects, enabling efficient mini-batch training and evaluation. The complete preprocessing workflow is visualized in Figure 1.

5.2 Training parameters

The fine-tuning process was carried out on Google Colab using an NVIDIA A100 GPU. To ensure computational efficiency without compromising accuracy, mixed-precision (FP16) training was enabled via the Hugging Face Accelerate library [Gugger et al., 2022]. The models were implemented using the PyTorch framework [Paszke et al., 2019] and the Transformers library [Wolf et al., 2020]. The models were trained for a maximum of 3 epochs. To prevent overfitting, an early stopping mechanism was employed; training was halted if the validation loss did not improve for 2 consecutive epochs. The input data was processed with a maximum sequence length of 384 tokens. Since many passages in SQuAD v2.0 exceed this limit, a sliding window approach was used with a document stride of 128 tokens, ensuring that answer spans located at the boundaries of truncated segments were not lost. For optimization, the AdamW algorithm [Loshchilov and Hutter, 2019] was selected due to its effective handling of weight decay and improved generalization capabilities in Transformer models. A linear learning rate scheduler was applied, utilizing a warmup period covering the first 10% of total training steps to stabilize the model parameters before the main optimization phase. The complete set of hyperparameters used for the models is detailed in Table 1.

Hyperparameter	**Value**	**Description**
Training Epochs (Max)	3	Maximum number of passes over the training data
Batch Size	16	Number of samples per gradient update
Learning Rate	2e-5	Initial learning rate for the optimizer
Optimizer	AdamW	Adaptive Moment Estimation with Weight Decay
Scheduler	Linear Decay	Linearly decays the learning rate after warmup
Warm-up Ratio	10%	Percentage of steps used for linear warmup
Precision Mode	FP16	Mixed precision training for memory efficiency
Early Stopping	2 epochs	Stops training if validation loss does not improve
Max Sequence Length	384	Maximum number of input tokens (with truncation)
Document Stride	128	Overlap size for sliding window on long contexts
Weight Decay	0.01	Regularization factor (default for AdamW)
Seed	42	Fixed random seed for reproducibility

Table 1: Training hyperparameters for the models

5.3 Training and optimization

All four models—ALBERT, RoBERTa, DeBERTa, and XLNet—were fine-tuned using cross-entropy loss, separately minimizing errors on start and end token predictions. The AdamW optimizer was selected for each architecture due to its robust handling of weight decay, which effectively mitigates overfitting in Transformer-based models. A linear learning rate decay schedule with warmup (10% of total steps) was consistently applied to stabilize training in the early stages while promoting convergence in later epochs. This unified training setup reflects practices widely adopted in state-of-the-art NLP research, ensuring that any observed differences in performance stem from the inherent architectural properties of the respective models (e.g., dynamic masking in RoBERTa, disentangled attention in DeBERTa, or permutation modeling in XLNet) rather than discrepancies in the training regimen.

5.4 Analysis of the data

Analyses of the SQuAD v2.0 dataset reveal structural patterns critical for model optimization. As illustrated in Figure 2, questions typically range between 10 and 20 words, while answers are predominantly concise, with the majority consisting of 1–5 words. A negligible correlation coefficient of 0.0018 between question and answer lengths confirms that answer complexity is independent of question length. Furthermore, the relative position analysis (Figure 3) indicates a distributional bias where answer spans appear more frequently at the beginning of context passages. These findings suggest that effective QA models must prioritize concise span extraction and remain robust against positional biases without ignoring later segments of the text.

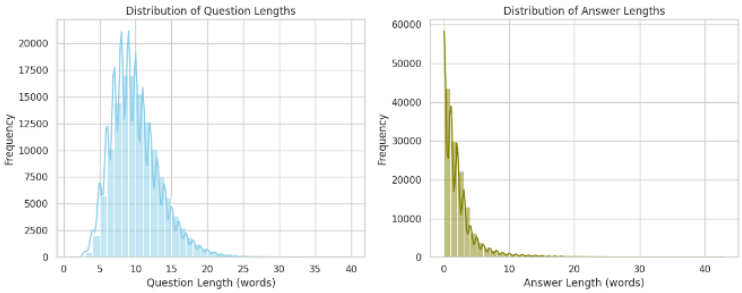

Figure 2: Length of questions and answers

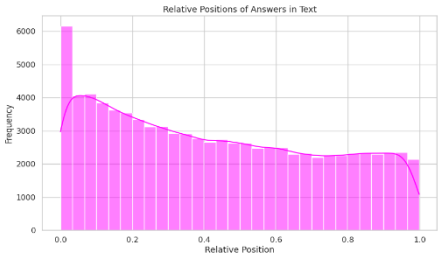

Figure 3: Relative positions of answers

5.5 Comparative results

In this study, the performance of ALBERT, RoBERTa, DeBERTa, and XLNet was comparatively analyzed on the SQuAD v2.0 dataset. All models were trained under identical conditions, using the same preprocessing steps, training parameters, and evaluation functions to ensure fairness and consistency. The evaluation focused on two widely adopted metrics in question answering tasks: Exact Match (EM), which measures the model's ability to extract the correct answer span word-for-word, and F1, which accounts for partial overlaps between predicted and ground-truth answers, providing a more flexible measure of accuracy [Rajpurkar et al., 2018]. To provide a holistic view of the results, Table 2 presents the performance metrics obtained in this study alongside those of other state-of-the-art models reported in the literature.

Model	**Reference**	**EM (%)**	**F1 (%)**
RoBERTa-Base	This Study	85.90	88.69
ALBERT-Base-V2	This Study	80.70	83.78
DeBERTa-V3-Base	This Study	85.41	88.54
XLNet	This Study	78.30	81.24
XLNet(Large)	[Yang et al., 2019]	87.90	90.60
RoBERTa-Large	[Liu et al., 2019]	89.40	94.60
DeBERTa-v3 (Large)	[He et al., 2023]	89.00	91.50
ALBERT-xxlarge	[Lan et al., 2020]	88.10	92.20
MiniLM	[Wang et al., 2020]	86.00	88.40

Table 2: Comparative evaluation results of the models

As detailed in Table 2, among the models trained in this study, RoBERTa-Base achieved the highest performance with 85.90% EM and 88.69% F1. DeBERTa-v3-Base followed closely, recording 85.41% EM and 88.54% F1, indicating that its disentangled attention mechanism offers robustness comparable to RoBERTa's dynamic masking strategy. XLNet-Base demonstrated competitive capability with 78.30% EM and 81.24% F1; while it trailed the autoencoding models in this specific setting, its permutation-based approach remains theoretically distinct. ALBERT-Base, while yielding lower absolute scores (80.70% EM and 83.78% F1), confirmed its trade-off design by maintaining respectable accuracy despite having significantly fewer parameters than its counterparts.

When viewed against the broader literature presented in Table 2, the obtained results indicate a competitive standing. While massive architectures such as XLNet-Large [Yang et al., 2019] and DeBERTa-v3-Large [He et al., 2023] achieve higher absolute scores (surpassing 90% F1), they require significantly more computational resources. Our RoBERTa-Base and DeBERTa-v3-Base implementations outperform efficiency-focused models like DistilBERT (82.47% F1) and standard BERT pipelines (79.37% F1). This comparison suggests that RoBERTa and DeBERTa offer an optimal balance for scenarios where "Large" models are computationally prohibitive. Conversely, ALBERT remains a highly viable option for strictly resource-constrained

environments where parameter efficiency is prioritized over maximizing raw accuracy [Lan et al., 2020].

Figure 4 illustrates the comparative EM and F1 performance of the evaluated transformer models on the SQuAD v2.0 dataset.

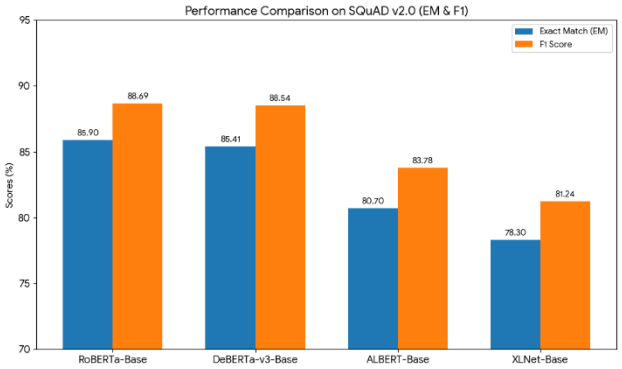

Figure 4: EM and F1 Scores of Models

Metric	**RoBERTa**	**ALBERT**	**DeBERTa**	**XLNet**
Total questions	11873	11873	11873	11873
EM on Answerable (%)	83.46	78.62	82.57	74.58
F1 on Answerable (%)	90.04	84.77	88.83	80.47
Total Answerable questions	5928	5928	5928	5928
EM on Unanswerable	87.35	82.77	88.24	82.00
F1 on Unanswerable	87.35	82.77	88.24	82.00
Total Unanswerable questions	5945	5945	5945	5945

Table 3: Performance of the evaluated models on answerable vs. unanswerable questions

As reported in Table 3, the detailed breakdown of results demonstrates that RoBERTa and DeBERTa exhibit superior robustness across both categories. RoBERTa achieved strong performance on answerable questions (83.46% EM / 90.04% F1), successfully identifying correct spans with high precision. DeBERTa followed closely in extraction tasks (82.57% EM / 88.83% F1) but notably outperformed all models on unanswerable questions with 88.24% EM/F1, compared to RoBERTa's 87.35%. This suggests that DeBERTa's disentangled attention mechanism is particularly effective at verifying constraints and abstaining when necessary.

In contrast, ALBERT and XLNet showed distinct limitations. ALBERT achieved respectable results on answerable questions (78.62% EM / 84.77% F1) but recorded lower scores on unanswerable ones (82.77%), indicating a higher tendency to generate false positives compared to the top-tier models. XLNet, while achieving consistent abstention rates (82.00%), lagged significantly in answer extraction (74.58% EM / 80.47% F1), suggesting that its permutation-based training might require more

extensive fine-tuning to master span boundary detection on this specific dataset. These findings confirm that performance on unanswerable questions is a decisive reliability factor; while RoBERTa and DeBERTa balance accurate predictions with correct abstentions, lighter or alternative architectures like ALBERT and XLNet trade off some of this robustness for efficiency or theoretical diversity.

Figures 5–8 illustrate the training and validation loss trajectories, along with validation F1 scores, for RoBERTa, ALBERT, DeBERTa, and XLNet, respectively.

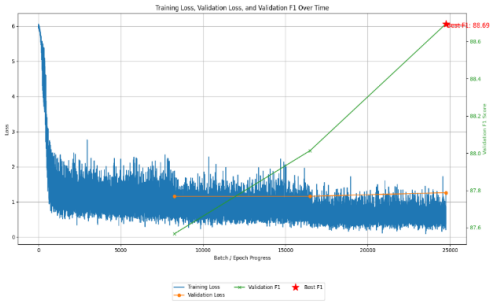

Figure 5: Training and Validation Loss of RoBERTa

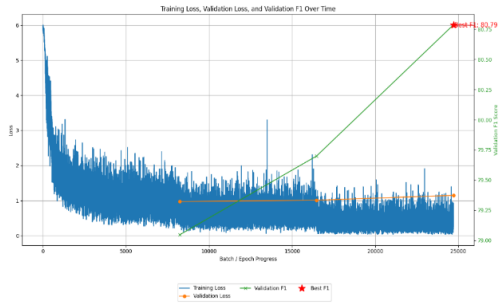

Figure 6: Training and Validation Loss of ALBERT

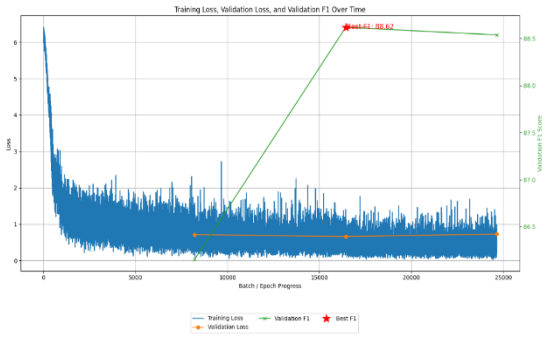

Figure 7: Training and Validation Loss of DeBERTa

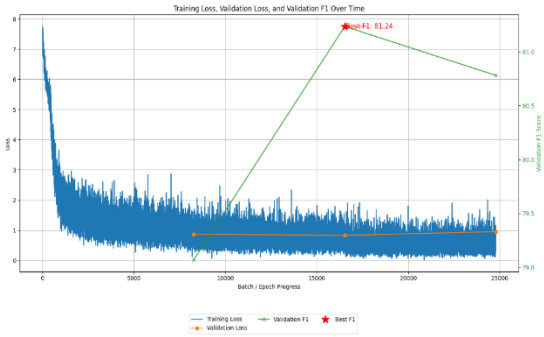

Figure 8: Training and Validation Loss of XLNet

The training and validation curves presented in Figures 5–8 illustrate the learning trajectories of RoBERTa, ALBERT, DeBERTa, and XLNet. Across all models, the training loss decreases rapidly during the initial phase, indicating that the pretrained representations are effectively leveraged during fine-tuning on the SQuAD v2.0 dataset. A distinct pattern emerged regarding the validation metrics: while the validation loss stabilized or showed slight fluctuations after the first epoch—a common phenomenon in fine-tuning large language models—the validation F1 scores continued to improve steadily until the end of training. This divergence suggests that the models continue to refine their span selection boundaries and decision thresholds even after the probabilistic cross-entropy loss saturated. The continuous rise in F1 scores confirms that none of the models suffered from destructive overfitting; instead, they successfully converged to their optimal performance states under the specified learning rate schedule and optimization constraints.

5.6 Qualitative error analysis

To investigate the limitations of the models beyond quantitative metrics, a qualitative error analysis was performed on RoBERTa's predictions. This analysis highlights specific failure modes, particularly regarding entity extraction and unanswerable questions. Table 4 presents representative examples of the observed error categories.

Question	**Answer**	**RoBERTa Prediction**	**Error Type**
Which of the Big Four teams did not lose a game in 2003-04?	Arsenal	Chelsea	Entity Distraction (Model retrieved a heavily featured entity in the same paragraph instead of the correct subject).

Why did Dyke only want the larger clubs on TV?	It would be more lucrative	Because of viewer preferences	Reasoning Error (Model selected a plausible-sounding semantic phrase rather than the causal explanation).
Who is the managing director of The Football League?	<No Answer>	Greg Dyke	False Positive (Model failed to abstain and extracted an entity associated with a similar role in the text).
What was proposed at the beginning of the 1991 season?	<No Answer>	<No Answer>	Correct Abstention (Model successfully identified that the context did not support an answer).

Table 4: Qualitative analysis of error types observed in RoBERTa predictions

The analysis reveals three primary failure modes. First, Entity Distraction occurs when the model correctly identifies the semantic category of the answer (e.g., a football club) but selects a salient entity appearing nearby in the context ("Chelsea") instead of the correct subject ("Arsenal"). This suggests a sensitivity to lexical frequency over syntactic dependency. Second, False Positives on Unanswerable Questions represent a significant challenge. For instance, when asked "Who is the managing director of The Football League?", the model incorrectly extracted "Greg Dyke," who was mentioned in the text with a similar job title, but for a different organization (LWT). This indicates that while the model is robust at extracting spans, it sometimes struggles to verify the precise constraints of the question against the context, leading to "hallucinated" answers for unanswerable queries. Finally, Reasoning Errors were observed in causal questions. In the case of why Dyke wanted larger clubs on TV, the model predicted "viewer preferences" instead of "lucrative," selecting a thematically related phrase rather than the direct cause. These findings highlight that while RoBERTa achieves high aggregate scores, future improvements should focus on strengthening the model's ability to verify

constraints in unanswerable scenarios and improving long-range dependency resolution to avoid distraction by salient entities.

5.7 Model complexity and training efficiency

The comparison of ALBERT, RoBERTa, DeBERTa, and XLNet highlights the critical trade-off between accuracy and computational efficiency in extractive question answering. ALBERT, utilizing parameter sharing and embedding factorization, reduces its parameter count to approximately 12 million. This design makes it highly memory-efficient and faster to train; in our experiments, it completed three epochs in approximately 39 minutes on an NVIDIA A100 GPU. ALBERT demonstrates suitability for resource-constrained environments where hardware limitations are critical, maintaining competitive performance despite its compact size.

In contrast, RoBERTa, DeBERTa, and XLNet represent the "Base" class of architectures with significantly higher complexity, each containing between 110 million and 125 million parameters. These models rely on large-scale pretraining and complex architectural features—such as dynamic masking in RoBERTa, disentangled attention in DeBERTa, and permutation modeling in XLNet. Consequently, their training incurs a higher computational cost, requiring approximately 75 to 80 minutes for three epochs under the same GPU settings. While this increased complexity translates into consistently higher EM and F1 scores (particularly for RoBERTa and DeBERTa), it comes at the expense of memory usage and training duration. In summary, while RoBERTa and DeBERTa achieve superior accuracy suitable for high-performance applications, ALBERT offers a resource-efficient alternative that balances reasonable performance with significantly lower hardware requirements.

6 Conclusions

This study investigated contextual question answering (QA) on the SQuAD v2.0 dataset, focusing on a comparative analysis of four distinct Transformer-based architectures: ALBERT, RoBERTa, DeBERTa, and XLNet. To ensure a fair assessment of their architectural capabilities—ranging from parameter efficiency and robust optimization to disentangled attention and permutation modeling—all models were trained and evaluated under identical preprocessing steps, hyperparameters, and evaluation protocols. The results provide a systematic perspective on the trade-offs between accuracy, computational efficiency, and robustness in handling unanswerable questions.

The experimental findings reveal distinct performance tiers. RoBERTa-Base achieved the highest overall performance (85.90% EM, 88.69% F1), validating the effectiveness of robust optimization strategies like dynamic masking and large-scale pretraining. DeBERTa-v3-Base followed closely (85.41% EM, 88.54% F1) and notably outperformed all other models in correctly abstaining from unanswerable questions (88.24%), highlighting the utility of its disentangled attention mechanism for verification tasks. XLNet-Base, while theoretically powerful with its permutation language modeling, yielded competitive but slightly lower results (78.30% EM, 81.24% F1) in this specific setting. ALBERT-Base, while trailing the larger architectures in raw accuracy (80.70% EM, 83.78% F1), confirmed its status as the

most resource-efficient model, offering a viable solution for memory-constrained environments with significantly fewer parameters.

A key insight derived from our qualitative error analysis is that performance on unanswerable questions is a decisive factor in overall model reliability. While RoBERTa and DeBERTa exhibited strong calibration in distinguishing answerable content from distractors, lighter or alternative architectures like ALBERT showed a higher tendency for "hallucinating" answers where none existed. This emphasizes that beyond raw span extraction, future QA models must focus on improving confidence calibration and abstention strategies to reduce false positives in real-world deployments.

The primary contribution of this study lies in its controlled, side-by-side evaluation of these diverse architectures. We demonstrate that model selection should be dictated by deployment priorities: when maximum accuracy and reliability are paramount, RoBERTa and DeBERTa are the preferable choices. Conversely, when efficiency and scalability are critical—such as in mobile applications or limited hardware settings—ALBERT offers a compelling alternative with only a moderate trade-off in accuracy.

Despite the comprehensive nature of this evaluation, the research has limitations. Experiments were restricted to the SQuAD v2.0 dataset and the English language; thus, generalizability across different domains and languages remains to be tested. Future work can extend this study by applying similar comparative frameworks to other NLP tasks (e.g., Named Entity Recognition), exploring ensemble strategies to combine the strengths of complementary architectures (e.g., RoBERTa's extraction with DeBERTa's verification), and examining performance in multilingual settings to address global deployment needs.

In summary, this study confirms that while RoBERTa remains the accuracy leader for SQuAD v2.0, DeBERTa offers superior robustness in unanswerable scenarios, and ALBERT provides an essential efficiency advantage. These findings reinforce the view that model selection for QA should not rely solely on headline scores, but rather on a nuanced balance between performance, efficiency, and the specific requirements of the application context.

References

[Chen et al. 2017] Chen, D., Fisch, A., Weston, J., Bordes, A.: "Reading Wikipedia to Answer Open-Domain Questions"; arXiv preprint arXiv:1704.00051 (2017).

[Clark et al. 2020] Clark, K., Luong, M. T., Le, Q. V., Manning, C. D.: "Electra: Pre-training Text Encoders as Discriminators Rather Than Generators"; arXiv preprint arXiv:2003.10555 (2020).

[Devlin et al. 2019] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding"; Proc. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019), 4171–4186.

[Gugger et al. 2022] Gugger, S., et al.: "Accelerate: Training and Deploying PyTorch Models with Ease"; GitHub Repository (2022) https://github.com/huggingface/accelerate.

[Harikrishnan et al. 2025] Harikrishnan, M., et al.:"Comparative Analysis of BERT Pipelines in SQuAD and MedQuAD Question Answering";in Challenges in Information, Communication and Computing Technology,Taylor & Francis / CRC Press, 2025.

[He et al. 2021] He, P., Liu, X., Gao, J., Chen, W.: "DeBERTa: Decoding-Enhanced BERT with Disentangled Attention"; Proc. Int. Conf. on Learning Representations (ICLR) (2021).

[He et al. 2023] He, P., Gao, J., Chen, W.: "DeBERTaV3: Improving DeBERTa Using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing"; Proc. Int. Conf. on Learning Representations (ICLR) (2023).

[Hochreiter and Schmidhuber 1997] Hochreiter, S., Schmidhuber, J.: "Long Short-Term Memory"; Neural Computation 9, 8 (1997), 1735–1780.

[Lan et al. 2020] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations"; arXiv preprint arXiv:1909.11942 (2019).

[Liu et al. 2019] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: "RoBERTa: A Robustly Optimized BERT Pretraining Approach"; arXiv preprint arXiv:1907.11692 (2019).

[Loshchilov and Hutter 2019] Loshchilov, I., Hutter, F.: "Decoupled Weight Decay Regularization"; Proc. Int. Conf. on Learning Representations (ICLR) (2019).

[Özkurt 2024] Özkurt, 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 (2024), 19–30. DOI: 10.69882/adba.chf.2024073

[Paszke et al. 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: "PyTorch: An Imperative Style, High-Performance Deep Learning Library"; Advances in Neural Information Processing Systems (NeurIPS) 32 (2019).

[Rajpurkar et al. 2016] Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: "SQuAD: 100,000+ Questions for Machine Comprehension of Text"; Proc. Conf. on Empirical Methods in Natural Language Processing (EMNLP) (2016), 2383–2392.

[Rajpurkar et al. 2018] Rajpurkar, P., Jia, R., Liang, P.: "Know What You Don't Know: Unanswerable Questions for SQuAD"; arXiv preprint arXiv:1806.03822 (2018).

[Sanh et al. 2019] Sanh, V., Debut, L., Chaumond, J., Wolf, T.: "DistilBERT, a Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter"; arXiv preprint arXiv:1910.01108 (2019).

[Vaswani et al. 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I.: "Attention Is All You Need"; Advances in Neural Information Processing Systems (NeurIPS) 30 (2017).

[Wang et al. 2020] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., Zhou, M.: "MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers"; Advances in Neural Information Processing Systems (NeurIPS) 33 (2020), 5776–5788.

[Wolf et al. 2020] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A.: "Transformers: State-of-the-Art Natural Language Processing"; Proc. 2020 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics (2020), 38–45.

[Yang et al. 2019] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., Le, Q. V.: "XLNet: Generalized Autoregressive Pretraining for Language Understanding"; Advances in Neural Information Processing Systems (NeurIPS) 32 (2019).

[Young et al. 2018] Young, T., Hazarika, D., Poria, S., Cambria, E.: "Recent Trends in Deep Learning Based Natural Language Processing"; IEEE Computational Intelligence Magazine 13, 3 (2018), 55–75.