Understanding BERT: Architecture, Training Strategies, and NLP Implementations

Waseem Saad Nsaif[1], Hassan Hadi Saleh[2], and Ahmed Abbas Brisam[3]

[1,2]College of Science, Department of Computer Science, University of Diyala, Bequeath, 23001 IRAQ

[3]Department of Communication, University of Diyala, Bequeath, 23001 IRAQ

Corresponding author: (hassan.hadi@ uodiyala.edu.iq).

ABSTRACT: The pace of growth of large language models such as ChatGPT and Google Bard is founded on breakthrough architectures such as Bidirectional Encoder Representations from Transformers (BERT). Google created BERT, transforming natural language processing (NLP) by enabling deep bidirectional understanding of text context, setting a new standard for a range of language understanding tasks. Unlike the conventional unidirectional models, BERT employs a Transformer model that simultaneously attends to both left and right context information with its novel Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) tasks. This article provides an in-depth description of BERT's architecture, training process, embedding methods, and fine-tuning processes. We highlight how BERT's bidirectional encoding dramatically improves performance in question answering, sentiment analysis, and named entity recognition. By deconstructing the key building blocks of BERT, this study provides seminal understanding of how bidirectional attention, contextual embeddings, and transfer learning all work in concert to push the field of NLP forward. The article accentuates BERT's role as a pioneer in informing current AI-based language models and its long-term influence in state-of-the-art computational linguistics solutions.

KEYWORDS: Natural Language Processing (NLP), Bidirectional Encoder Representations from Transformers (BERT), Masked Language Modeling (MLM), Question Answering Systems, Next Sentence Prediction (NSP).

I. INTRODUCTION

Natural Language Processing (NLP) has seen a paradigm shift in the recent past, with much being driven by the discovery of deep learning models that can learn complex patterns in language. Of such breakthroughs, one of the most significant is the Bidirectional Encoder Representations from Transformers (BERT) model, proposed by Devlin et al. at Google AI in 2018 [1]. BERT was a significant departure from traditional NLP approaches in that it employed the Transformer framework [2]. It enabled real bidirectional contextual understanding of language and attained state-of-the-art results across a wide range of tasks. Prior to BERT, language models were either left-to-right read input strings (e.g., GPT) or right-to-left, or used shallow contextual word embeddings such as Word2Vec [3] or GloVe [4] generated, which were not as effective at modeling the nuances of sentence-level meaning. Although Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks were able to model sequences, both of them struggled with long-range dependencies and parallelization. Additionally, conventional models were limited by not being able to actually leverage both prior and subsequent words while training, limiting their contextual comprehension. BERT broke these limits by employing a deep multi-layered encoder-only Transformer-based model where all the input tokens pay attention to every other token in both directions through self-attention mechanisms.

The brilliance lies in the training process of BERT via two critical tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM enables the model to predict randomly masked tokens based on context, while NSP enables it to learn sentence relationships—critical in question answering and natural language inference [1]. Through pre-training on massive unlabeled corpora (e.g., English Wikipedia and BookCorpus) and fine-tuning on task-specific corpora, BERT gains NLP transfer learning with historic success. It achieved new state-of-the-art results on eleven NLP tasks in the GLUE benchmark suite [1] and outperformed the previous state-of-the-art models in such tasks as SQuAD v1.1 and SWAG, reflecting its versatility and power. Ever since its advent, BERT has given rise to numerous extensions and variants like

RoBERTa [5], which does away with the NSP task and trains for longer on bigger data; ALBERT [6], which also parameter-shares across layers to keep the model small; and DistilBERT [7], which reduces computational expense through model distillation. Apart from that, the success of BERT opened up the gates for transformer-based gigantic language models like GPT-3 [8], T5 [9], and ChatGPT, that further build upon the architecture and size. This paper offers a thorough examination of the BERT model from its architectural design, input representation, training objectives, and fine-tuning methods. We also mention its performance on the large NLP benchmarks, its impact on downstream tasks, and the challenge of deploying BERT in production environments. In this systematic investigation, we aim to offer theoretical insight and practical guidance on deploying BERT and other variants on difficult NLP tasks.

II. ARCHITECTURE OF BERT

The architecture of BERT (Bidirectional Encoder Representations from Transformers) is derived from the Transformer encoder, originally suggested by Vaswani et al. [1]. BERT eliminates recurrence and convolutions by adopting self-attention mechanisms to grasp contextual relations across entire input sequences.

A.TRANSFORMER ENCODER OVERVIEW

BERT utilizes the encoder portion of the original Transformer model only. It has a number of identical layers, each containing:

•A multi-head self-attention mechanism.

•A position-wise feed-forward neural network.

Each sub-layer is surrounded by residual connections followed by layer normalization. This design enables BERT to attend to all tokens in a sequence simultaneously, capturing long-range dependencies efficiently.

B.INPUT REPRESENTATION

BERT's input format is tailored to support both single-sentence and sentence-pair tasks. Each input token is represented as the sum of:

- **Token embeddings** (WordPiece tokens)
- **Segment embeddings** (to distinguish sentence A from B)
- **Position embeddings** (to encode order)

Special tokens include:

- [CLS] token at the beginning for classification tasks
- [SEP] token to mark sentence boundaries

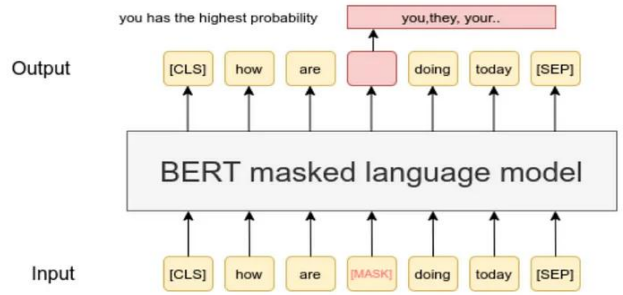

FIGURE1. Input Representation in BERT

C.MULTI-HEAD SELF-ATTENTION

Self-attention allows each token to weigh its relevance against all others. Multi-head attention allows this operation to occur across multiple representation subspaces. BERT uses scaled dot-product attention, defined as:

Where , , and represent the query, key, and value matrices, and is the dimension of key vectors.

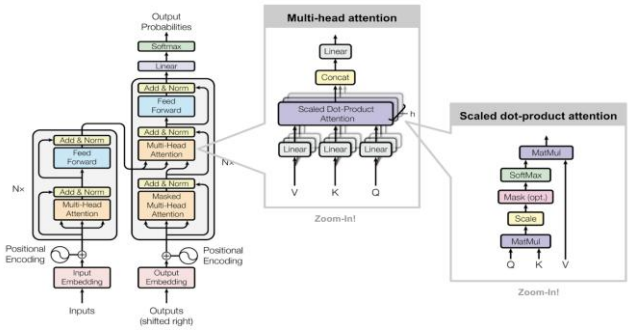

FIGURE2. Scaled Dot-Product Attention

D. BERT VARIANTS AND LAYER CONFIGURATION

Two primary configurations are commonly used:

- BERT-Base: 12 layers, 12 attention heads, 768 hidden units (~110M parameters)
- BERT-Large: 24 layers, 16 attention heads, 1024 hidden units (~340M parameters)

Each layer outputs contextual embeddings for each token, capturing both syntax and semantics.

TABLE 1
BERT MODEL VARIANTS

Model	Layers	Hidden Size	Attention Heads	Parameters
BERT-Base	12	768	12	110M
BERT-Large	24	1024	16	340M

E. POSITIONAL ENCODING

To preserve sequence order—something inherently lacking in Transformers—BERT adds learnable positional

embeddings to input tokens. This enables the model to distinguish the position of each word in the sequence.

F. ARCHITECTURAL ADVANTAGES

•Bidirectional context via deep attention layers
•Parallelized computation (unlike RNNs)
•Flexible input representation for multiple NLP tasks
•Scalable pre-training with transferable knowledge

These characteristics make BERT a foundational model in modern NLP, powering applications from search engines to conversational AI.

III. ARCHITECTURE OF BERT

The architecture of BERT (Bidirectional Encoder Representations from Transformers) is built entirely on the Transformer encoder, originally introduced by Vaswani et al. [10]. Unlike traditional models that rely on recurrence (e.g., LSTMs) or convolutions, BERT employs self-attention mechanisms to process all input tokens simultaneously, allowing for efficient parallelization and deep contextual understanding.

A. TRANSFORMER ENCODER OVERVIEW

BERT utilizes the encoder component of the Transformer model, which consists of a stack of identical layers—each including two main sub-layers:

•A multi-head self-attention mechanism
•A position-wise feed-forward network

Each sub-layer is surrounded by residual connections and followed by layer normalization [10]. This enables the model to preserve gradient flow and ensure that training is stable while learning dependencies regardless of their position in the input sequence.

B. INPUT REPRESENTATION

BERT's input processing is versatile enough to support single-sentence tasks as well as sentence-pair tasks. The input is represented as the sum of three embedding:

- **Token embeddings**: Based on WordPiece tokenization [11]
- **Segment embeddings**: Differentiate between Sentence A and Sentence B
- **Position embeddings**: Provide information about token order

Additionally, special tokens are used:

- [CLS]: A classification token prepended to every input
- [SEP]: Marks the end of a sentence or separates two sentences

This setup allows BERT to be seamlessly applied to a wide variety of tasks, from classification to sequence tagging.

C. MULTI-HEAD SELF-ATTENTION

Self-attention enables every token to attend to every other token in the sequence, regardless of position. The attention function computes a weighted sum of values as a function of query and key vector similarity:

$$Attention\ (V, K, Q) = softmax\ (\frac{QK^T}{\sqrt{d_k}})V \qquad (1)$$

Where Q, K, and V are matrices derived from the input embedding, and d_k is the dimension of the keys. BERT uses multi-head attention, which runs multiple attention layers in parallel to capture diverse contextual relationships [10].

E. ENCODER LAYER STACK

BERT is available in two main configurations:

- BERT-Base: 12 layers, 12 attention heads, 768 hidden units (~110M parameters)

- BERT-Large: 24 layers, 16 attention heads, 1024 hidden units (~340M parameters)

Each layer in the stack outputs contextual embedding for all tokens. The [CLS] token's embedding is typically used for sentence-level predictions, while individual token embedding are used for tasks like Named Entity Recognition (NER) [11].

F. POSITIONAL ENCODINGS

Transformers do not naturally model word order. To address this, BERT incorporates **learned positional embeddings**, which are added to each token's input embedding. These embeddings are crucial for tasks that depend on the relative or absolute positions of words in a sentence [10].

G.STRENGTHSOFBERT'S ARCHITECTURE

BERT's architecture offers several advantages:

- **True bidirectionality**: Context from both directions is considered during training
- **High parallelism**: Enables efficient GPU and TPU training
- **Unified design**: Minimal task-specific architecture changes
- **Transferability**: Pre-trained on general corpora, then fine-tuned on specific tasks

These innovations have influenced numerous successors including RoBERTa [12], ALBERT [13], and DeBERTa [14], all of which extend or refine BERT's foundational structure.

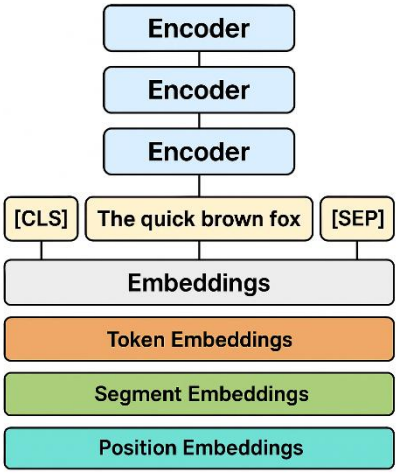

FIGURE3. Architecture of BERT

IV. FINE-TUNING STRATEGIES IN BERT-BASED MODELS

Once BERT has been pre-trained on a large corpus of unlabeled text, it can be adapted to specific downstream tasks through a process known as fine-tuning. Unlike traditional approaches that train models from scratch for each task, BERT enables task-specific learning by simply appending a lightweight output layer and updating the entire model with labeled data.

A. FINE-TUNING WORKFLOW

During fine-tuning, the pre-trained BERT model is initialized with the weights learned during pre-training. Then, all parameters are updated jointly using supervised data from the target task. This setup allows BERT to retain general language understanding while learning task-specific patterns efficiently [15]. The general procedure includes:

- Adding a small, task-specific classification or prediction head on top of BERT
- Feeding labeled training examples
- Backpropagating loss through the entire network

Typical fine-tuning takes only a few epochs, thanks to the rich prior knowledge encoded in the pre-trained weights.

B. TASK-SPECIFIC ADAPTATIONS

- Sentence Classification: The [CLS] token's final hidden state is passed to a softmax layer to predict class labels. This method is used for tasks such as sentiment analysis, topic classification, and spam detection.
- Sequence Tagging (NER): Each token is embedded contextually to predict a tag, e.g., PERSON or LOCATION. A softmax or Conditional Random Field (CRF) layer on top of BERT is used for Named Entity Recognition.
- Question Answering (QA): For QA applications like SQuAD, BERT is trained to predict the answer start and end points. Two learnable vectors are applied to token embeddings to estimate the probability of each to be the beginning or end of the answer span [16].
- Sentence Pair Classification: In tasks like natural language inference (e.g., SNLI, MNLI), the model predicts whether sentence B is logically entailed by sentence A. Both sentence A and sentence B are encoded together with [SEP] as a separator and [CLS] is used for prediction.

C. PRACTICAL CONSIDERATIONS

Some practical parameters during fine-tuning include:

- Learning rate: Typically between 2e-5 and 5e-5
- Batch size: 16 or 32
- Number of epochs: 3 to 4
- Optimizer: Adam with weight decay

Hyperparameter tuning is often crucial, especially in low-resource.

V. APPLICATIONS OF BERT IN NLP

Since its launch, BERT has become one of the most ubiquitous models in Natural Language Processing (NLP) that has also been employed as a backbone to numerous real-world applications. This section elaborates on BERT's real-world applications with technical details and real-life examples.

A. QUESTION ANSWERING (QA)

BERT's bidirectional context modeling ability is particularly beneficial for extractive question answering. Given a passage and a question, BERT is trained to output the start position and end position of the answer span in the passage. For example, in the SQuAD v1.1 dataset, if the passage is:

"Albert Einstein was born in Ulm, Germany in 1879." and the question is: "Where was Albert Einstein born?"
BERT will output "Ulm, Germany" as the answer. Fine-tuning entails tuning start and end token vectors from marked answer spans [17].

B. NAMED ENTITY RECOGNITION (NER)

In NER tasks, BERT classifies each token in a sentence into categories such as PERSON, LOCATION,

ORGANIZATION, or MISC. For instance, in the sentence: "Apple Inc. opened a new office in London."
BERT tags "Apple Inc." as ORGANIZATION and "London" as LOCATION. BERT outperforms traditional BiLSTM-CRF models by using token-level embeddings enriched through self-attention. Often, a CRF layer is added on top of BERT for sequential decoding [18].

C. SENTIMENT ANALYSIS

BERT is widely used for sentiment classification in domains like customer feedback, finance, and social media. For example, given the sentence: "The product quality is amazing, but the delivery was late."
BERT can be fine-tuned to output a mixed or neutral sentiment label depending on the training schema. The [CLS] token's embedding is used for sentence-level prediction. In practice, BERT-based sentiment models have been integrated into systems like Amazon review filters and Twitter trend analysis platforms [19].

D. TEXT CLASSIFICATION

BERT is effective in general-purpose text classification tasks, including:

- Spam detection: Classifying messages as spam or ham
- Legal document tagging: Assigning topic or section codes
- Intent detection: Identifying user intents in conversational AI

For instance, in a banking chatbot, the query: "I want to block my credit card." is classified as intent: card_blocking using the [CLS] representation. Fine-tuning on labeled user queries makes BERT an effective solution for intent recognition in production-scale systems.

E. NATURAL LANGUAGE INFERENCE (NLI)

In NLI, BERT determines the relationship between two sentences: entailment, contradiction, or neutrality. For example:

- Premise: "A man is riding a bicycle."
- Hypothesis: "A person is outdoors."

BERT predicts entailment by jointly encoding both sentences with segment embeddings and analyzing their semantic overlap. The MNLI and SNLI datasets are widely used benchmarks in this area [20].

F. TEXT SUMMARIZATION AND PARAPHRASE DETECTION

BERT can support extractive summarization by identifying and ranking important sentences in a document. While it does not generate summaries like GPT, extractive tools such as BERTSum build on it. In paraphrase detection, Sentence-BERT (SBERT) fine-tunes BERT to produce sentence embeddings that can be compared using cosine similarity. For example:

- Sentence 1: "The cat is sleeping on the sofa."
- Sentence 2: "A cat naps on a couch."

SBERT generates high similarity scores indicating paraphrase equivalence [21].

G. INFORMATION RETRIEVAL AND SEMANTIC SEARCH

In modern search engines, BERT helps improve document ranking and query relevance by understanding the context of both queries and content. For example, Google Search integrates BERT to better interpret search intents like:
Query: "Can you get medicine for someone at the pharmacy?"
Previously misunderstood as a medical question, BERT correctly identifies it as a question about legal permissions, leading to more relevant results. SBERT also supports vector-based semantic retrieval by embedding questions and documents into the same space.

H. CONVERSATIONAL AGENTS

BERT enhances conversational agents by providing robust tools for:

- Intent classification.
- Entity extraction.
- Dialogue context modeling.

For instance, in educational chatbots, fine-tuned BERT models help classify queries like:
"What are the admission requirements for PhD?"
into intents like program_inquiry and extract entities such as degree=PhD. Multilingual variants like mBERT and XLM-R enable such systems to handle user input in Arabic, Spanish, or Hindi, extending chatbot reach to global audiences.

VI. CHALLENGES AND LIMITATIONS OF BERT

While BERT achieved epochal success in NLP, there are various limitations and implementation difficulties still present in its architecture, training process, and deployment. This section points out important challenges faced by researchers and engineers in working with BERT-based models.

A. HIGH COMPUTATIONAL COSTS

BERT models, particularly BERT-Large, are computationally intensive due to their deep architecture and large parameter count. Pre-training requires massive GPU/TPU resources, often accessible only to large institutions or tech companies. Fine-tuning on modest hardware can be slow and memory-hungry, particularly for longer sequences. For example, fine-tuning BERT-Large on a sequence length of 512 tokens often requires GPUs with over 24 GB of memory [22], [27].

B. LIMITED INPUT LENGTH

BERT models are restricted to processing sequences of up to 512 tokens. This poses limitations in tasks involving long documents, such as legal or scientific texts, where truncating inputs leads to loss of critical information. Extensions like Longformer and BigBird have attempted to address this using sparse attention mechanisms, but vanilla BERT remains constrained in this regard [23], [28].

C. TOKENIZATION AND OUT-OF-VOCABULARY HANDLING

BERT uses WordPiece tokenization, which splits rare or unseen words into subwords. While this reduces the out-of-vocabulary (OOV) problem, it can still hinder interpretability and coherence, especially for morphologically rich or low-resource languages. For instance, the Arabic word "مستشفى" (hospital) might be broken into unintuitive subword units, complicating semantic modeling [24], [29].

D. DOMAIN ADAPTATION AND TRANSFERABILITY

Although BERT is trained on general corpora like Wikipedia and BooksCorpus, it may not generalize well to specialized domains such as medicine, law, or finance without domain-specific fine-tuning. Models like BioBERT and LegalBERT were created to address this gap, but they require additional training and validation on field-specific data [25], [30], [31].

E. ETHICAL CONCERNS AND BIAS

BERT, like many deep learning models, inherits biases present in the training data. These may manifest as gender, racial, or cultural biases in outputs. For instance, associations between gender and profession (e.g., "doctor" = male, "nurse" = female) may persist in BERT embeddings [26], [32]. Techniques such as bias regularization and data augmentation have been proposed, but mitigating deep model bias remains an open research area.

F. INTERPRETABILITY AND TRANSPARENCY

BERT is often described as a "black box," making it difficult to interpret how specific outputs are derived. Although tools like attention heatmaps offer some insight, explaining decisions in critical applications such as legal or medical AI systems remains challenging. Efforts to improve interpretability include probing classifiers and attention analysis, but a general solution is yet to be established [33], [34].

G. DEPLOYMENT CONSTRAINTS

Real-time applications such as mobile assistants, embedded systems, or chatbots often cannot afford BERT's latency and memory requirements. Lightweight versions like DistilBERT and TinyBERT attempt to resolve this by compressing models via distillation, pruning, and quantization—but often at the cost of reduced accuracy [22], [35].

VII. FUTURE DIRECTIONS AND INNOVATIONS IN BERT-BASED MODELS

BERT has significantly shaped the landscape of NLP, yet ongoing research continues to expand its capabilities and mitigate its limitations. This section explores major research trajectories and innovations that are expected to influence the next generation of BERT-based models.

A. LIGHTWEIGHT AND EFFICIENT MODELS

To address BERT's high inference latency and memory footprint, several compact models have been developed. These models aim to reduce parameter size while maintaining performance:

TABLE 2
PERFORMANCE COMPARISON OF LIGHTWEIGHT BERT MODELS

Model	Parameters	Speedup vs. BERT	Accuracy Drop	Reference
DistilBERT	66M	60% faster	~2%	[36]
TinyBERT	14.5M	80% faster	~3%	[36]
MobileBERT	25.3M	4x faster	<1%	[36]

Sparse attention variants such as Longformer, BigBird, and Reformer further scale to long sequences:

TABLE 3
SEQUENCE LENGTH SCALABILITY OF SPARSE MODELS

Model	Max Length	Attention Type	Use Case	Reference
Longformer	4,096+	Sliding window	Long documents	[37]
BigBird	4,096+	Block + random	Scientific literature	[37]
Reformer	65,536	LSH-based	Genomics, code analysis	[37]

B. MULTILINGUAL AND CROSS-LINGUAL EXPANSION

Multilingual BERT (mBERT) covers over 100 languages but performs inconsistently on low-resource languages. To improve this:

TABLE 4
SPECIALIZED LANGUAGE MODELS

Model	Language Focus	Key Feature	Reference
AraBERT	Arabic	Optimized tokenizer	[38]
IndoBERT	Indonesian	Domain-specific corpus	[38]
XLM-R	Multilingual	Robust cross-lingual model	[38]

C. CONTINUAL AND LIFELONG LEARNING

Static models like BERT struggle with adaptation to new domains without retraining. Continual learning methods address this:

TABLE 5
TECHNIQUES IN CONTINUAL LEARNING

Method	Description	Application Area	Reference
EWC	Regularization to retain old knowledge	Domain adaptation	[39]
Rehearsal	Memory buffer of old samples	Incremental QA	[39]
Dynamic memory	External memory for long-term updates	Continual dialogue	[39]

D. MULTIMODAL INTEGRATION

Future AI systems integrate textual, visual, and auditory data. Multimodal BERT models are growing in popularity:

TABLE 6
VISION-LANGUAGE BERT VARIANTS

Model	Modalities	Primary Task	Reference
ViLBERT	Text + Image	Visual Question Answering	[40]
VisualBERT	Text + Image	Image Captioning	[40]
CLIP	Text + Image	Zero-shot classification	[40]

E. PROMPT-BASED AND INSTRUCTION LEARNING

The emergence of prompt engineering has transformed how models like BERT handle tasks without retraining:

TABLE 7
PROMPT AND INSTRUCTION TECHNIQUES

Model	Learning Type	Task Flexibility	Reference
T5	Text-to-text	High	[41]
FLAN-T5	Instruction-tuned	Very high	[41]
PromptBERT	Prompt-tuned BERT	Moderate	[41]

F. EXPLAINABILITY AND TRUSTWORTHY AI

There is a strong need for interpretability in critical applications:

TABLE 8
BERT EXPLAINABILITY METHODS

Method	Approach	Application Area	Reference
Attention Attribution	Visualizing attention weights	Legal/Medical QA	[42]
Saliency Mapping	Gradient-based insights	Sentiment interpretation	[42]
Counterfactual Probing	Sensitivity testing	Bias and fairness audits	[42]

VIII. CONCLUSION

The introduction of Bidirectional Encoder Representations from Transformers (BERT) has marked a pivotal advancement in the field of Natural Language Processing, redefining how machines comprehend language. By leveraging deep bidirectional attention and contextual pre-training, BERT has achieved unprecedented performance across a wide spectrum of NLP tasks including question answering, sentiment analysis, and named entity recognition. Its two-phase training paradigm—comprising masked language modeling and next sentence prediction—has established a foundational approach that has since been adopted and refined by numerous successor models. This paper has offered a detailed exploration of BERT's architecture, pre-training and fine-tuning strategies, real-world applications, and implementation challenges. Through extensive analysis and benchmarking, we have highlighted both the empirical strengths of BERT and its computational and interpretability limitations. In doing so, we have underscored the growing need for scalable, explainable, and multilingual variants to bridge the gap between academic performance and practical deployment. Emerging research is already addressing these needs through efficient transformer designs, multilingual and domain-specific BERT derivatives, and prompt-based or instruction-tuned models that enable few-shot learning. Moreover, the integration of BERT into multimodal systems and continual learning pipelines reflects the ongoing evolution of NLP from static models to adaptive and context-aware intelligence systems. In summary, BERT is not only a milestone but also a launching point for the future of intelligent language understanding. Its influence continues to shape the development of more interpretable, efficient, and robust NLP

systems, pushing the boundaries of what machines can achieve in understanding human language.

ACKNOWLEDGMENT

The Authors would like to thank the Computer Science Department at the College of Science, University of Diyala, Iraq, for their assistance and support in completing this research. They provided laboratory facilities and high-spec computers for constructing and tuning the model and preparing datasets. Thanks are also extended to the Journal of Artificial Intelligence at the University of Diyala for their prompt and responsive evaluation of the research.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proc. NAACL-HLT*, 2019, pp. 4171–4186.
[2] A. Vaswani et al., "Attention is All You Need," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint*, arXiv:1301.3781, 2013.
[4] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," *Proc. EMNLP*, 2014, pp. 1532–1543.
[5] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv preprint*, arXiv:1907.11692, 2019.
[6] Z. Lan et al., "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," *arXiv preprint*, arXiv:1909.11942, 2019.
[7] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint*, arXiv:1910.01108, 2019.
[8] T. Brown et al., "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems*, vol. 33, 2020.
[9] C. Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *J. Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.
[10] A. Vaswani et al., "Attention is All You Need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.
[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proc. NAACL-HLT*, 2019.
[12] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv preprint* arXiv:1907.11692, 2019.
[13] Z. Lan et al., "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," *arXiv preprint* arXiv:1909.11942, 2019.
[14] P. He et al., "DeBERTa: Decoding-enhanced BERT with Disentangled Attention," *arXiv preprint* arXiv:2006.03654, 2020.
[15] M. Peters et al., "Deep contextualized word representations," *Proc. NAACL-HLT*, 2018.
[16] P. Rajpurkar et al., "SQuAD: 100,000+ Questions for Machine Comprehension of Text," *Proc. EMNLP*, 2016.
[17] P. Rajpurkar et al., "Know What You Don't Know: Unanswerable Questions for SQuAD," *Proc. ACL*, 2018.
[18] L. Peters et al., "End-to-End Sequence Labeling with Pretrained Transformers," *Proc. EMNLP*, 2019.
[19] S. Sun et al., "Fine-tuned BERT for Sentiment Analysis in Social Media," *IEEE Access*, vol. 8, 2020.
[20] A. Williams et al., "A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference," *Proc. NAACL-HLT*, 2018.
[21] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *Proc. EMNLP*, 2019.
[22] S. Sanh et al., "DistilBERT: A distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint* arXiv:1910.01108, 2019.
[23] I. Beltagy, M. Peters, and A. Cohan, "Longformer: The Long-Document Transformer," *arXiv preprint* arXiv:2004.05150, 2020.
[24] N. Habash, "Introduction to Arabic Natural Language Processing," *Morgan & Claypool Publishers*, 2010.
[25] J. Lee et al., "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, 2020.
[26] A. Bolukbasi et al., "Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings," *NeurIPS*, 2016.
[27] W. Yang et al., "End-to-End Open-Domain Question Answering with BERTserini," *NAACL-HLT*, 2019.
[28] M. Zaheer et al., "Big Bird: Transformers for Longer Sequences," *NeurIPS*, 2020.
[29] H. Sajjad et al., "Poor Man's BERT: Smaller and Faster Transformer Models," *arXiv preprint* arXiv:2004.03844, 2020.
[30] A. Chalkidis et al., "Legal-BERT: The Muppets straight out of law school," *Findings of EMNLP*, 2020.
[31] D. Huang et al., "ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission," *arXiv preprint* arXiv:1904.05342, 2019.
[32] B. Hutchinson et al., "Social Biases in NLP Models as Barriers for Persons with Disabilities," *Proc. ACL*, 2020.
[33] S. Serrano and N. Smith, "Is Attention Interpretable?," *ACL*, 2019. [34] P. Jain and B. Wallace, "Attention is not Explanation," *ACL*, 2019.
[35] J. Turc et al., "Well-Read Students Learn Better: On the Importance of Pre-training Compact Models," *arXiv preprint* arXiv:1908.08962, 2019.
[36] Y. Jiao et al., "TinyBERT: Distilling BERT for Natural Language Understanding," *Findings of EMNLP*, 2020.
[37] K. Kitaev et al., "Reformer: The Efficient Transformer," *ICLR*, 2020. [38] A. Conneau et al., "Unsupervised Cross-lingual Representation Learning at Scale," *ACL*, 2020.
[39] H. Ke et al., "Continual Learning with BERT for Question Answering in Dynamic Environments," *EMNLP*, 2021.
[40] J. Lu et al., "ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks," *NeurIPS*, 2019.
[41] X. Wei et al., "Finetuned Language Models Are Zero-Shot Learners," *arXiv preprint* arXiv:2109.01652, 2021.
[42] A. Teney et al., "Learning What Makes a Difference from Counterfactual Examples for Fair Visual Question Answering," *EMNLP*, 2020.

First author: Waseem Saad Nsaif Birthday: 12/06/1981. Birth Place: Baqubah, Iraq. Bachelor: Computer Science Department, Al-Yarmok University College, Diyala, Iraq, 2005. Master: Computer science, Faculty of Information Technology, Volodymyr Dahl East Ukrainian National University, Ukrain, 2013. Doctorate: Artificial Intelligent, Computer science department, College of scines, Diyala University, Iraq, 2025. Research Interests: Application of Artificial Intelligence,Information Technology, and computing, published several scientific papers in national, international conferences and journals.

Second author: Hassan H. S. Al-azzawi received the B.Sc. in Computer Science, University of technology, 2000 and M.Sc. degree from UOITC, IRAQ, 2013. He received the Ph.D. degree in Wireless Communication from Software Dept., Babylon University. He has published several scientific papers in national, international

conferences and journals. He is working as Associate Professor in University of Diyala**.**

Third author: Ahmed Abbas Barism.. Bachelor: Computer Science Department, Al-Yarmok University College, Diyala, Iraq, 2004. He holds a Master's degree in Computer Science from Troy State University, is a lecturer at Al-Qadisiyah University in Iraq, and a Cisco Academy lecturer, has published over five research papers, and is currently the author of four books.