A Research on DistilBERT and ALBERT: Advances in Transformer-Based NLP Models

Raghad Mohamed

September, 2025

Abstract

Both DistilBERT and ALBERT are improvements built on top of the BERT model, a pre-trained transformer-based language model introduced by Google AI in 2018. BERT was trained on large-scale text data (English Wikipedia and BookCorpus) and marked a turning point in Natural Language Processing (NLP).

DistilBERT is a compressed version of BERT, trained using *knowledge distillation*. It manages to retain around 97% of BERT's performance while being 40% smaller and up to 60% faster, making it suitable for real-world applications on resource-limited devices such as mobile phones.

ALBERT, proposed by Google Research in 2019, focuses on parameter efficiency. It introduces techniques like factorized embeddings, cross-layer parameter sharing, and Sentence Order Prediction (SOP) loss. These innovations drastically reduce the number of parameters while maintaining or even improving performance.

This paper provides an overview of BERT, then explores the motivation, design, and contributions of DistilBERT and ALBERT.

I. Introduction

What is BERT in simple terms?

BERT (Bidirectional Encoder Representations from Transformers) is a deep learning model designed to understand language by looking at the context of a word from both directions (left and right). This bidirectional understanding allows BERT to outperform earlier models like Word2Vec, GloVe, and LSTMs on many NLP tasks.

Why were DistilBERT and ALBERT developed?

While BERT achieved state-of-the-art results, it is extremely large, with hundreds of millions of parameters. This made it slow and expensive to train and deploy. DistilBERT and ALBERT were developed to solve this issue from different angles. DistilBERT reduces model size and speeds up inference using knowledge distillation. ALBERT improves training efficiency and reduces memory consumption with parameter sharing and embedding factorization. Both models show that high accuracy can be achieved while balancing resource usage.

II. Background

NLP before BERT

Earlier models like Word2Vec and GloVe represented words as vectors but struggled with context, “bank” in “river bank” vs. “bank account” would be treated the same. Recurrent Neural Networks (RNNs) and LSTMs improved sequence modeling but were *slow* and had *difficulty with long dependencies*.

The Transformer architecture

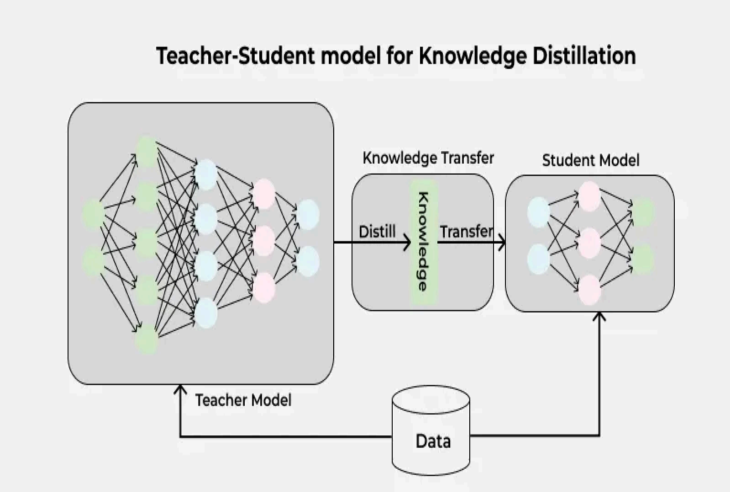
The Transformer, introduced in 2017, relies on attention mechanisms rather than recurrence. This allows for parallelization and better handling of long-range dependencies. BERT is based on this architecture.

III. BERT Model Architecture

BERT introduces several key innovations that set it apart from earlier models. First, it is bidirectional, meaning it looks at the context of a word from both directions, rather than just left-to-right or right-to-left. It also uses a Masked Language Model (MLM), where random words are masked during training, and the model learns to predict them using surrounding context. In addition, BERT employs Next Sentence Prediction (NSP) to capture relationships between sentences. Finally, BERT comes in two main sizes: BERT-Base, with 110 million parameters, and BERT-Large, with 345 million parameters, illustrating how increasing model size can significantly impact performance.

IV. DistilBERT and ALBERT

DistilBERT



To train the model, researchers used the same data as BERT (Wikipedia + BookCorpus), but with a more efficient training process. DistilBERT was built using **Knowledge Distillation**, where a smaller "student" model is trained to mimic a larger "teacher" model (BERT). Its architecture is nearly identical to BERT but with fewer layers. Specifically, token-type embeddings and the pooler are removed.

DistilBERT training uses a combination of **three different losses**:

1. **Distillation Loss:** the student learns from the teacher's probability distribution over classes.

$$L_{ce} = -\sum_{i} t_i \log(s_i)$$

where ti is the teacher's predicted probability for class i, and si is the student's probability.

2. **Masked Language Modeling Loss**
   Same as in BERT, where some tokens are masked, and the model learns to predict them.

$$L_{mlm} = -\sum_{i \in M} \log P(x_i \mid x_{\setminus M})$$

3. **Cosine Embedding Loss**
   - Aligns the hidden states of the student and teacher to ensure similar representations.

$$L_{cos} = 1 - \cos(h^{(s)}, h^{(t)})$$

The final objective is a **linear combination** of all three losses:

$$L = \alpha L_{ce} + \beta L_{mlm} + \gamma L_{cos}$$

DistilBERT achieved remarkable performance: it retained ~97% of BERT's accuracy while being **40% smaller** and up to **60% faster**.

Applications: It is widely used in question answering, sentiment analysis, and text classification, especially in resource-constrained environments like mobile devices.

ALBERT

ALBERT was introduced in 2019 as a more efficient variant of BERT, focusing on reducing model size without sacrificing accuracy. Instead of compressing through distillation, ALBERT rethinks BERT's architecture.

1. Factorized Embedding Parameterization
   it separates input embeddings (context-independent) from hidden embeddings (context-dependent). This reduces the number of parameters in the embedding layer by up to 80% while maintaining representational power.
2. Cross-Layer Parameter Sharing
   Instead of learning separate parameters for each transformer layer, ALBERT shares weights across layers. This drastically reduces redundancy, leading to a 70% reduction in the total parameter count.
3. Sentence Order Prediction (SOP)
   ALBERT replaces BERT's Next Sentence Prediction (NSP) with SOP. While NSP mixes topic prediction with coherence, SOP focuses purely on sentence order and coherence. This makes the model better at handling tasks that require understanding the logical flow of text.

Impact:
By combining these techniques, ALBERT achieves performance comparable to or even better than BERT, but with **far fewer parameters**, making it more scalable and resource-friendly.

Applications:
Like DistilBERT, ALBERT excels in tasks such as question answering, sentiment analysis, and natural language inference. Its efficiency makes it especially useful for large-scale deployments where memory and compute are limited.

V. Applications of BERT and Its Variants

BERT and its variants have been applied to a wide range of real-world tasks. One major use is **sentiment analysis**, where the model detects opinions in text, such as identifying whether a review is positive or negative. Another important application is **question answering**, which powers chatbots and search engines by allowing them to understand and respond to user queries more effectively. BERT is also widely used in **Named Entity Recognition (NER)**, where the goal is to identify and classify names, dates, and locations within text. Perhaps most famously, **Google Search** integrates BERT to better understand the intent behind user queries and deliver more accurate results.

VI. Limitations and Future Directions

Despite their success, BERT and its variants still face important limitations. One challenge is the **computational cost**, since training requires powerful GPUs or TPUs, making it resource-intensive. Another issue is the **data requirement**, as pre-training depends on massive text corpora that are not always easy to obtain or process. Over time, **successors such as RoBERTa, DistilBERT, and ALBERT** have pushed the boundaries further, while even newer GPT-style transformers continue to redefine what is possible in NLP. Looking ahead, the **future outlook** of this field is centered on making models smaller, faster, and more specialized, while still maintaining high accuracy and strong performance across tasks.

VII. Conclusion

BERT *revolutionized* NLP by introducing a deep contextual understanding of language. DistilBERT and ALBERT built on this foundation to address efficiency challenges, DistilBERT through compression, and ALBERT through parameter optimization. These models prove that powerful NLP systems can be both effective and efficient, paving the way for wider adoption in real-world applications.

References

- Sanh et al., 2019. *DistilBERT: A Distilled Version of BERT* (arXiv:1910.01108)
- Lan et al., 2019. *ALBERT: A Lite BERT for Self-Supervised Learning of Language Representations* (arXiv:1909.11942)
- DistilBERT in Natural Language Processing – GeeksforGeeks
- ALBERT: A Light BERT for Supervised Learning – GeeksforGeeks
- (PDF) *The DistilBERT Model: A Promising Approach to Improve Machine Reading Comprehension Model*