Efficient Variants of BERT: DistilBERT and ALBERT

Youssef Salah

Abstract

This report discusses two efficient variants of BERT: DistilBERT and ALBERT. The motivation behind these models is reducing computational cost and memory usage while maintaining strong language understanding performance.

Contents

1 Introduction

1.1 What is BERT

BERT (Bidirectional Encoder Representations from Transformers), introduced by Google in 2018, is a language model that understands text by looking at both left and right context at the same time. It uses the Transformer encoder architecture and it is pretrained on two tasks. First, Masked Language Modeling (predicting missing words) and Next Sentence Prediction (understanding sentence relationships and ordering). Due to pretraining on large text corpora, BERT can be fine-tuned for specific tasks like sentiment analysis, question answering or named entity recognition, achieving strong performance across many benchmarks.

1.2 Advantages and Breakthroughs of BERT:

- Bidirectional Context Understanding improving word understanding.
- Using Transformer based architecture to capture long range dependencies in text.
- Versatility across multiple tasks for example question answering, named entity recognition etc.

Devlin et al., 2018.

1.3 Variants of BERT

While BERT achieves excellent performance, it is large and computationally expensive, making training and deployment costly. Variants of BERT have been developed to reduce model size, speed up inference, or improve efficiency without sacrificing much accuracy.

Popular Variants:

- DistilBERT
- ALBERT
- RoBERTa
- TinyBERT
- Electra
- SpanBERT

2 DistilBERT

DistilBERT, introduced by Sanh et al. in 2019, is a smaller, faster, and lighter version of BERT. It is designed to retain most of BERT's language understanding capabilities while reducing computational cost and memory usage, making it suitable for deployment in resource-constrained environments.

2.1 Core Idea

DistilBERT uses **knowledge distillation**, a teacher-student training approach. The original BERT model acts as the teacher, and DistilBERT (the student) learns to mimic its outputs. The model is trained to match the teacher's predicted distributions and hidden states, allowing it to capture most of BERT's knowledge in fewer parameters.

2.2 Architecture Differences

Compared to BERT, DistilBERT has several modifications:

- **Reduced number of layers:** 6 Transformer layers instead of 12.
- **No token-type embeddings:** Simplifies input representation.
- **No Next Sentence Prediction (NSP):** Focuses only on Masked Language Modeling.
- **Smaller model size:** Approximately 40% fewer parameters than BERT.

2.3 Training Objective

DistilBERT combines multiple loss components during training:

$$\mathcal{L} = \alpha\mathcal{L}_{MLM} + \beta\mathcal{L}_{KD} + \gamma\mathcal{L}_{cos} \tag{1}$$

where:

- $\mathcal{L}_{MLM}$ is the standard Masked Language Modeling loss.
- $\mathcal{L}_{KD}$ is the distillation loss comparing student and teacher logits.
- $\mathcal{L}_{cos}$ matches the teacher's hidden states using cosine similarity.

This objective allows DistilBERT to efficiently learn both output predictions and intermediate representations.

2.4 Performance and Trade-offs

DistilBERT achieves a strong balance between speed and accuracy:

- **Speed:** Around 60% faster inference than BERT.
- **Memory:** 40% fewer parameters, lower VRAM usage.
- **Accuracy:** Retains roughly 97% of BERT's performance on GLUE benchmarks.

The model is ideal for applications where latency or hardware constraints are critical, such as mobile or edge devices.

3 ALBERT

ALBERT (A Lite BERT), introduced by Lan et al. in 2019, is an efficient variant of BERT designed to reduce the number of parameters while maintaining performance. It achieves this through **cross-layer parameter sharing** and **factorized embedding parameterization**, making it more memory-efficient and scalable to larger models.

3.1 Core Idea

The main goals of ALBERT are:

- **Parameter reduction:** Decrease model size without reducing hidden layer dimensions.
- **Improved training efficiency:** Reuse parameters across layers to reduce memory footprint.

3.2 Factorized Embedding Parameterization

ALBERT separates the large vocabulary embedding matrix into two smaller matrices:

$$E = E_{vocab \times d_{emb}} \cdot E_{d_{emb} \times d_{hidden}} \tag{2}$$

where $d_{emb} \ll d_{hidden}$. This reduces the number of parameters in the input embeddings while keeping the hidden size unchanged.

3.3 Cross-Layer Parameter Sharing

Instead of using unique parameters for each Transformer layer, ALBERT shares weights across all layers:

- The same parameters are reused in each layer, significantly reducing model size.
- This strategy also stabilizes training and improves scaling behavior for very deep models.

3.4 Sentence Order Prediction (SOP)

ALBERT replaces BERT's Next Sentence Prediction (NSP) with **Sentence Order Prediction**:

- SOP predicts whether two consecutive sentences are in the correct order.
- This task encourages the model to capture inter-sentence coherence, improving performance on downstream tasks.

3.5 Performance and Trade-offs

- **Parameter efficiency:** ALBERT significantly reduces parameters compared to BERT (e.g., ALBERT-xxlarge has 18× fewer parameters than BERT-large).
- **Strong performance:** Achieves competitive results on GLUE, SQuAD, and other NLP benchmarks.
- **Scalability:** Cross-layer sharing allows for training deeper models without memory bottlenecks.

ALBERT is particularly suitable for tasks where model size and memory efficiency are critical, without sacrificing accuracy.

4 Comparison: DistilBERT vs ALBERT

Feature	**DistilBERT**	**ALBERT**
Compression Method	Knowledge Distillation	Parameter Sharing + Factorize
Layers Reduced	Yes (6 vs 12)	No (shares weights acros
Parameter Sharing	No	Yes (cross-layer)
Embedding Factorization	No	Yes (reduces embedding pa
Training Objective	KD Loss + MLM	MLM + SOP
Model Size Reduction	∼40% fewer params	Up to 18× fewer params than
Inference Speed	∼60% faster	Comparable to BERT (slightly faster
Targeted Advantage	Faster inference, smaller memory	Parameter efficiency, scalability

Table 1: Technical comparison between DistilBERT and ALBERT.

5 Conclusion

DistilBERT and ALBERT are efficient variants of BERT that address its computational and memory limitations. DistilBERT focuses on faster inference and smaller memory footprint using knowledge distillation, while ALBERT achieves significant parameter reduction through cross-layer sharing and factorized embeddings. Both models retain strong language understanding performance, making them suitable for resource-constrained environments and large-scale NLP applications.

References

- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* https://arxiv.org/abs/1810.04805
- Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter.* https://arxiv.org/abs/1910.01108
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). *ALBERT: A Lite BERT for Self-supervised Learning of Language Representations.* https://arxiv.org/abs/1909.11942