# ALBERT for claim verification task

Jianyu (Fisher) Shi
W266: Natural Language Processing
UC Berkeley School of Information
fisher.shi@ischool.berkeley.edu

## Abstract

Fact Extraction and VERification (FEVER) (Thorne et al., 2018a) is a purpose-built dataset for claim verification against textual sources. This project explores the potential of using A Lite BERT (ALBERT) (Lan et al., 2020) for the FEVER claim verification task, and compares its performance against the traditional BERT model. ALBERT outperformed BERT in many NLP benchmarks with shorter training time. However, for the specific implementation on the FEVER dataset, it had worse performance with no training (fine-tuning) speed advantage.

## 1. Introduction

The abundance of information online has not only increased information accessibility to everyone, but has also accelerated the spread of unverified/inaccurate information around the planet. This is a critical problem in today's world because exposure to unverified/inaccurate information can cause attitudes of inefficacy, alienation, and cynicism toward a certain group of the population in the society.

Claim verification problem has received a lot of attention in the NLP community. While a lot of progress has been made in the recent years, there is always a need for a better, faster model. One good candidate is ALBERT. ALBERT showed impressive results on many NLP benchmarks with less training time compared with BERT, or Bidirectional Encoder Representations from Transformers (Devlin et al., 2018), However, there was little effort to apply ALBERT on the claim verification problems. The purpose of this project is to explore the potential of using ALBERT on Claim Verification tasks.

## 2. Background

### 2.1 Dataset

Fact Extraction and VERification (FEVER) (Thorne et al., 2018a) is a purpose-built dataset for claim verification against textual sources. The dataset contains 185,445 human-generated

claims, manually verified against the introductory sections of Wikipedia pages and labeled as SUPPORTED, REFUTED or NOT_ENOUGH_INFO. For the first two classes, systems and annotators need to also return the combination of sentences forming the necessary evidence supporting or refuting the claim (Figure 1).

| |
|---|
| **Claim:** Roman Atwood is a content creator.<br>**Evidence:** **`[wiki/Roman_Atwood]`** He is best known for his vlogs, where he posts updates about his life on a daily basis.<br>**Verdict:** SUPPORTED |
| **Claim:** Furia is adapted from a short story by Anna Politkovskaya.<br>**Evidence:** **`[wiki/Furia_(film)]`** Furia is a 1999 French romantic drama film directed by Alexandre Aja, who co-wrote screenplay with Grgory Levasseur, adapted from the science fiction short story Graffiti by Julio Cortzar.<br>**Verdict:** REFUTED |
| **Claim:** Afghanistan is the source of the Kushan dynasty.<br>**Verdict:** NOT ENOUGH INFO |

Figure 1: Manually verified claim requiring evidence from multiple Wikipedia pages.

The dataset was partitioned into training, development, and test sets. The dev and test sets have balanced samples for each class (Table 1). It was ensured that each Wikipedia page used to generate claims occurs in exactly one set.

| Split | SUPPORTED | REFUTED | NEI |
|---|---|---|---|
| Training | 80,035 | 29,775 | 35,639 |
| Dev | 3,333 | 3,333 | 3,333 |
| Test | 3,333 | 3,333 | 3,333 |
| Reserved | 6,666 | 6,666 | 6,666 |

Table 1: Dataset split sizes for SUPPORTED, REFUTED and NOT_ENOUGH_INFO (NEI) classes

## 2.2 Evaluation Metrics

The official evaluation metrics for the FEVER claim verification task include Label Accuracy (LA) and FEVER score. LA is a general evaluation metric, which calculates claim classification accuracy rate without considering retrieved evidence. FEVER score is the label accuracy conditioned on providing at least one complete set of evidence. Sentences correctly labeled as NOT_ENOUGH_INFO do not require evidence. Where multiple sets of evidence was annotated in the data, only one set was required for the claim to be considered correct for the FEVER score.

## 2.3 Baseline

### 2.3.1 Random Baseline

Given that the development and test datasets have balanced class distributions, a random baseline will have 33% Label Accuracy if one ignores the requirement for evidence for SUPPORTED and REFUTED.

### 2.3.2 FEVER Baseline

In the original FEVER paper, the authors developed a pipeline approach which, given a claim, first identifies relevant documents (Document Retrieval), then selects sentences forming the evidence from the documents (Sentence Retrieval) and finally classifies the claim w.r.t. Evidence (Claim Verification). The original FEVER paper achieves 31.87% accuracy in verification when requiring correct evidence to be retrieved for claims SUPPORTED or REFUTED, and 50.91% if the correctness of the evidence is ignored.

## 2.4 Related Works

Existing fact verification models usually employ FEVER's three-step pipeline system: document retrieval, sentence retrieval and claim verification.

### 2.3.1 Document Retrieval

Most previous works reuse the document retrieval component of top-performing systems (Hanselowski et al., 2018b; Yoneda et al., 2018; Nie et al., 2019) in the FEVER Shared Task challenge (Thorne et al., 2018b), which treated the Document Retrieval step as an entity linking problem (Cucerzan, 2007) and employs the constituency parser from AllenNLP (Gardner et al., 2017). After parsing the claim, every noun phrase is considered as a potential entity mention. Then it uses these phrases as queries to find relevant Wikipedia pages through the online MediaWiki API. This approach achieved 93.55% accuracy for document retrieval.

| #search results | doc. accuracy |
|---|---|
| 3 | 92.60 |
| 5 | 93.30 |
| 7 | **93.55** |

Table 2: Performance of the retrieval systems achieved by (Hanselowski et al., 2018b) using different numbers of MediaWiki search results

### 2.3.2 Sentence Retrieval

With some small modifications, Enhanced Sequential Inference Model (ESIM) (Chen et al., 2016) has been used in (Nie et al., 2019; Hanselowski et al., 2018) for Sentence Retrieval. ESIM encodes premises and hypotheses using one Bidirectional Long Short-Term Memory (BiLSTM) with shared weights. The encoded sentences are later aligned by a bidirectional attention mechanism. The encoded and aligned sentences are combined, and another shared BiLSTM matches the two representations. Finally, a softmax layer classifies the max and mean pooled representations of the second BiLSTM.

BERT is another popular approach for Sentence Retrieval. A softmax layer is added on the last hidden state of the first token for the Sequence Classification task with the claim-evidence-pair as the input.

### 2.3.3 Claim Verification

GEAR (Zhou et al., 2019) formulates claim verification as a graph reasoning task and provides two kinds of attention. It conducts reasoning and aggregation over claim evidence pairs with a graph model. BERT has also been widely used in the Claim Verification step and achieved great performance (Li et al., 2019; Soleimani et al., 2019). Zhong et al.(2019) further employs XLNet (Yang et al., 2019) and establishes a semantic-level graph for reasoning for better performance.

# 3. ALBERT for Sentence Retrieval and Claim Verification

Just as for many other NLP applications, BERT and its variance such as XLNet (Yang et al., 2019) and RoBERTa (Liu et al., 2019) has achieved great results for both Sentence Retrieval and Claim Verification. However, one major drawback of BERT is the size of the model. With hundreds of millions of parameters, BERT is expensive to train and also takes a lot of memory.

Lan et al., 2020 proposed A Lite BERT (ALBERT) architecture, which has significantly fewer parameters than a traditional BERT architecture.

| Model | | Parameters | Layers | Hidden | Embedding | Parameter-sharing |
|---|---|---|---|---|---|---|
| BERT | base | 108M | 12 | 768 | 768 | False |
| | large | 334M | 24 | 1024 | 1024 | False |
| ALBERT | base | 12M | 12 | 768 | 128 | True |
| | large | 18M | 24 | 1024 | 128 | True |
| | xlarge | 60M | 24 | 2048 | 128 | True |
| | xxlarge | 235M | 12 | 4096 | 128 | True |

Table 3: Comparison of the model size and number of parameters between BERT and ALBERT

While with much fewer parameters, ALBERT (with its ALBERT-xxlarge configuration) has achieves both an overall 30% parameter reduction compared to the BERT-large model, and performance gains over many NLP benchmarks such as SQuAD (Rajpurkar et al., 2016) and RACE (Lai et al., 2017).

| Model | | Parameters | SQuAD1.1 | SQuAD2.0 | MNLI | SST-2 | RACE | Avg | Speedup |
|---|---|---|---|---|---|---|---|---|---|
| BERT | base | 108M | 90.4/83.2 | 80.4/77.6 | 84.5 | 92.8 | 68.2 | 82.3 | 4.7x |
| | large | 334M | 92.2/85.5 | 85.0/82.2 | 86.6 | 93.0 | 73.9 | 85.2 | 1.0 |
| ALBERT | base | 12M | 89.3/82.3 | 80.0/77.1 | 81.6 | 90.3 | 64.0 | 80.1 | 5.6x |
| | large | 18M | 90.6/83.9 | 82.3/79.4 | 83.5 | 91.7 | 68.5 | 82.4 | 1.7x |
| | xlarge | 60M | 92.5/86.1 | 86.1/83.1 | 86.4 | 92.4 | 74.8 | 85.5 | 0.6x |
| | xxlarge | 235M | **94.1/88.3** | **88.1/85.1** | **88.0** | **95.2** | **82.3** | **88.7** | 0.3x |

Table 4: Comparison of the performance on NLP benchmarks between BERT and ALBERT

It appears that ALBERT is a faster and better model than BERT. If so, how would ALBERT perform on the FEVER dataset?

To answer this question, I fine-tuned the pre-trained model of BERT-base and ALBERT-base from HuggingFaceon on both the Sentence Retrieval and Claim Verification, using the state-of-art Document Retrieval results provided by UKP-Athene, which includes all the predicted Wiki articles associated with the claims.

For the Sentence Retrieval step, all of the sentences in the introductory paragraph of the predicted articles are paired with the corresponding claims. All the sentence pairs were then assigned a label of whether the sentence is the actual evidence of the claim. Then the sentence pairs, together with the label, were tokenized in the form of [CLS] + sentence 1 + [SEP] + sentence 2. Then the model was trained as a binary Sequence Classification task. The outcome is the top 5 sentences as predicted evidence for each claim.

Similarly, for the Claim Verification step, the top 5 predicted pieces of evidence were paired with their corresponding claims. All the sentence pairs were then assigned a label of [SUPPORT], [REFUTE], or [NOT_ENOUGH_INFO] and then tokenized. The model was then trained as a 3-class Sequence Classification task. A simple majority vote of the top 5 evidence for each claim was then implemented to produce the final prediction.

In terms of the hyperparameters, for each step, I used a batch size of 32, learning rate of 2e-5, and 2 epochs.

# 4. Results and discussion

## 4.1 The Results

The table below shows the performance of ALBERT-base is worse than BERT-base on any metrics. For Sentence Retrieval, the Evidence Recall (the most important metric in the sentence

retrieval step as we want to capture as many true positives as possible) of BERT-base was 0.7045, while the Evidence Recall of ALBERT-base was only 0.6072. For the final FEVER score, BERT achieved 0.6898 while ALBERT only got 0.6054.

| Model Name | FEVER Score | Label Accuracy | Evidence Precision | Evidence Recall | Evidence F1 |
|---|---|---|---|---|---|
| BERT | 0.6898 | 0.7407 | 0.8931 | 0.7045 | 0.7877 |
| ALBERT | 0.6054 | 0.6564 | 0.859 | 0.6072 | 0.7115 |

Table 5: Comparison of performance on FEVER dev dataset between BERT-base and ALBERT-base

Not only did ALBERT-base not out-perform BERT-base on the FEVER dataset, there was no noticeable difference on the training (fine-tuning) and inference time between the two models, despite the fact that ALBERT-base model appears to have only 1/9 of the parameters (12M vs 108M).

## 4.2 Potential Explanations

ALBERT has a unique concept called “Parameter Sharing”, which means the parameters used in each encoder layer are identical. For example, for ALBERT-base, the parameters of any of the 12 layers are identical to those in other layers. In other words, ALBERT-base has only 1 encoder layer that applies 12 times. In contrast, the parameters of the 12 layers of BERT are all unique. This helps explain the faster training for ALBERT when training the model from scratch. However, when fine-tuning the models on a custom dataset, the model needs to iterate through the same number of layers as BERT. Therefore, Parameter Sharing does not help reduce the training time for model fine-tuning.

The Parameter Sharing feature of ALBERT also helps explain the performance drop when comparing with the BERT model with the same number of the hidden layers. Since all of the 12 hidden layers of ALBERT-base are identical to each other, it appears that ALBERT-base learns less context information about the training data compared with BERT, which has 12 unique encoder layers to learn the features.

Based on the findings, it appears that for model fine-tuning, BERT is still a better choice compared with ALBERT. If that’s the case, it seriously limits the potential of using ALBERT on more use cases, especially for anyone with limited resources to train the models from scratch. However, it would be interesting to explore the potential of training ALBERT from scratch for a future project and compare its performance with BERT.

# References

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018a. FEVER: a large-scale dataset for fact extraction and VERification. In Proceedings of NAACL, pages 809–819.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017a. Reading wikipedia to answer opendomain questions. In Proceedings of ACL, pages 1870–1879.

Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018b. UKP-athene: Multisentence textual entailment for claim verification. In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER). Association for Computational Linguistics.

Takuma Yoneda, Jeff Mitchell, Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. UCL machine reading group: Four factor framework for fact finding (HexaF). In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER).

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. Proceedings of the AAAI Conference on Artificial Intelligence.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018b. The fact extraction and VERification (FEVER) shared task. In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER).

Silviu Cucerzan. 2007. Large-Scale Named Entity Disambiguation Based on Wikipedia Data. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLPCoNLL), pages 708–716, Prague, Czech Republic.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. AllenNLP: A deep semantic natural language processing platform.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, and Hui Jiang. 2016. Enhancing and combining sequential and tree LSTM for natural language inference. CoRR, abs/1609.06038.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. CoRR, abs/1810.04805.

Jackson Luken, Nanjiang Jiang, and Marie-Catherine de Marneffe. 2018. QED: A fact verification system for the fever shared task. In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pages 156–160.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. GEAR: Graph-based evidence aggregating and reasoning for fact verification. In Proceedings of ACL, pages 892–901.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In Proceedings of NeurIPS, pages 5754–5764.

Tianda Li, Xiaodan Zhu, Quan Liu, Qian Chen, Zhigang Chen, and Si Wei. 2019. Several experiments on investigating pretraining and knowledge enhanced models for natural language inference. arXiv preprint arXiv:1904.12104.

Amir Soleimani, Christof Monz, and Marcel Worring. 2019. BERT for evidence retrieval and claim verification. arXiv preprint arXiv:1910.02655.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In International Conference on Learning Representations.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. RACE: Large-scale ReAding comprehension dataset from examinations. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 785–794, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1082.

UKPLab / FEVER-2018-team-athene GitHub: https://github.com/UKPLab/fever-2018-team-athene

HuggingFace pretrained models: https://huggingface.co/transformers/pretrained_models.html