COMP90042 Natural Language Processing Project: Evidence Retrieval and Claim Classification for Unverified Scientific Claims

Group 13

Abstract

This report presents a two-task NLP system for detecting and classifying unverified scientific claims. It retrieves the top-200 evidence using SVD-reduced Word2Vec embeddings, re-ranks them via a Transformer, and classifies claims using an attention-enhanced Transformer. Our report describes the performance of different models under different hyper-parameter settings. The report concludes with an analysis of evaluation methods and experimental details.

1 Introduction

Climate change presents a serious threat to humanity in recent decades. Despite the overwhelming consensus among scientists, certain individuals and interest groups continue to spread misinformation and unfounded claims. Unverified scientific claims are disseminated at an unprecedented pace across today's highly interconnected information networks. These unverified scientific claims have distorted public opinion and eroded public confidence in the scientific community. To address these challenges, our group plans to design an NLP and machine learning based filter that recognizes and flags unverified claims before they permeate public discourse.

Our system consists of the following two parts: Evidence Retrieval: This module is designed to identify and retrieve the most relevant scientific evidence from a structured knowledge base based on a given claim. Claim Classification: This module is designed to classify the retrieved evidence into 4 different labels: SUPPORTS, REFUTES, NOT ENOUGH INFO, DISPUTED. This report begins by outlining our experimental process in searching for the most effective model architecture. It then focuses on the implementation of pre-trained Word2Vec embeddings (Google News) and a Transformer-based model for evidence retrieval and claim classification. Various model parameters were fine-tuned to optimize performance and improve accuracy across both tasks.

2 Data Preprocessing

To focus on informative keywords and ensure consistent model input, all claim and evidence sentences undergo a three-step text preprocessing: token lemmatization, stop word removal, and punctuation filtering.

3 Model and Embedding Selection

3.1 Baseline Setup

To investigate the impact of different sequence modeling approaches on the performance of the text matching task, we first established a baseline model without semantic embeddings. This baseline leverages TF-IDF for text vectorization and applies cosine similarity to retrieve relevant evidence, because of its simplicity and computational efficiency (Nehra, 2024). Then we evaluate the below three classical sequence models: GRU, LSTM and Bidirectional LSTM.

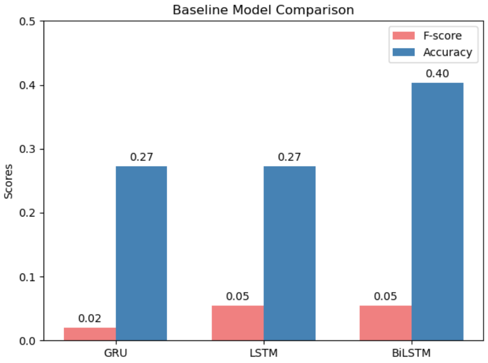

Figure 1: Three Baseline Model Comparison

Based on the results shown in Figure 1, the Bidirectional LSTM achieves the highest classification accuracy among the three models under the Top-3 evidence setting. Therefore, we adopt the **TF-IDF**

+ Bidirectional LSTM combination as the baseline for subsequent comparisons.

3.2 Embedding Selection

We explore Transformer-based embeddings to further enhance semantic representation. Specifically, we experiment with two pretrained models for semantic embedding. The first is **BERT** (bert-base-uncased) from Hugging Face Transformers, which is used as a **frozen embedding extractor**, meaning that the BERT parameters remain fixed during training. This approach eliminates the need for additional text preprocessing or tokenization, as the BERT tokenizer automatically handles lowercasing and WordPiece segmentation (huggingface.co, n.d.). The second is the **Word2Vec embedding**, pretrained on the Google News corpus, which provides 300-dimensional static word vectors (Google, 2019).

Although the BERT and Word2Vec embeddings are different, we use the same parameters for both Transformer Encoder. Specifically, we set the feed-forward hidden dimension to 256, the number of encoder layers to 2, the number of attention heads to 4, and apply no dropout. This allows us to compare the performance of different embeddings under the consistent settings.

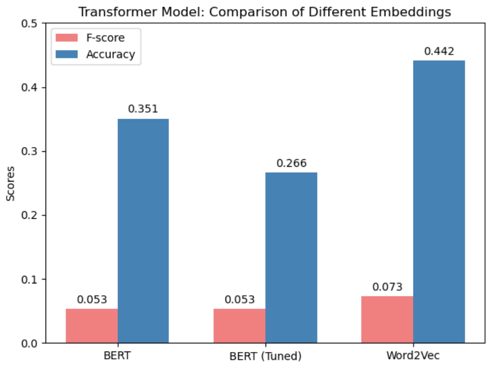

Figure 2: Different Embeddings based on Transformer

From the above Figure 2, we observe that the performance of Word2Vec embeddings is obviously better than the performance of BERT embeddings under the same Transformer Model settings. Then we try to fine-tune the Transformer Model with BERT embedding to investigate whether the performance will be better. We increase the feedforward hidden dimension to 512 and the number of encoder layers to 3. Additionally, we apply the dropout rate of 0.3 to mitigate overfitting. But the performance of BERT embeddings after fine-tune became even worse. So, we suppose that Word2Vec embeddings have a higher degree of fitness in this topic.

The Word2Vec embeddings can provide stable and fixed dimensional word vectors, which has obvious advantages in similarity computation and fast matching of large-scale candidate evidence. In contrast, although BERT embeddings capture dynamic semantic variations across different contexts, they tend to be less stable, which may lead to overfitting or performance fluctuations in certain semantic retrieval tasks (Reimers and Gurevych, 2019). Therefore, we do not consider BERT embeddings in our final model and instead select **Word2Vec embeddings** combined with the Transformer encoder.

3.3 Compare Word2Vec with Transformer & Word2Vec with Bidirectional LSTM

Based on previous experiments, we find that the Bidirectional LSTM achieves the best performance among three baseline models, while Word2Vec is the most effective embedding method. Hence, we combine Bidirectional LSTM with Word2Vec embedding and compare its performance against the Transformer Model using the same embedding.

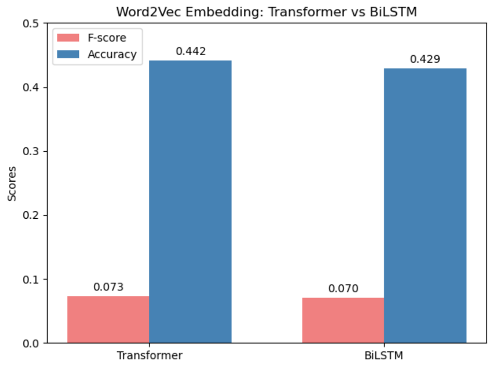

Figure 3: Transformer Vs Bidirectional LSTM

From the above Figure 3, we observe that the performance of the Transformer model with Word2Vec embeddings is better than that of the Bidirectional LSTM with the same embeddings. Notably, this performance is achieved without any parameter tuning of the Transformer model. So, we prefer to choose the Transformer model.

Although bidirectional LSTM can capture both forward and backward contextual information, this model still struggles to model long-range dependencies effectively. For this project, there are 1208827 candidate evidences in total. The bidirectional LSTM processes each piece of evidence sequentially due to its recurrent structure, which leads to computational inefficiency when handling large-scale datasets. Furthermore, as the number of candidate evidence increases, the model's ability to retain relevant information decreases, which may lead to vanishing gradients (Hochreiter and Schmidhuber, 1997). So, it becomes more difficult to identify and align the most relevant evidence with the claim.

However, the Transformer model effectively compensates for the limitations of bidirectional LSTM. With its self-attention mechanism, all positions in the sequence can be processed in parallel, which significantly increases computationally efficiency. Moreover, the transformer is not prone to the vanishing gradient problem, as it is non-recurrent by design and incorporates layer normalization (Vaswani et al., 2017). Therefore, we obviously adopt the **transformer model** with **Word2Vec embeddings**.

4 Methodology

4.1 Initial Idea

In our initial exploration, we found that Word2Vec embeddings provided a better semantic representation of dataset tokens than Bert-base. Based on this insight, we built a backbone framework using Transformer encoders and introduced a two-stage evidence retrieval pipeline to improve both retrieval accuracy and computational efficiency. All experiments applied early stopping to reduce the risk of overfitting by halting training once the validation loss stopped improving.

4.1.1 Task 1: Evidence Retrieval (ER)

The evidence retrieval process consists of two stages. In the pre-ranking stage, we computed the cosine similarity between SVD-reduced Word2Vec embeddings (128-dimensional) of each claim and the entire evidence corpus (approximately one million paragraphs), retrieving the top-6 to top-8 most relevant candidates. In the subsequent re-ranking stage, a lightweight 3-layer Transformer encoder (with hidden dimension 256 and dropout 0.3) was used to score the semantic relevance between the claim and each candidate. This re-ranking step replaced the initial cosine-based scoring and enabled the model to learn richer claim-evidence relationships. The top-K ranked evidence were then selected as inputs for the next classification task.

4.1.2 Task 2: Claim Classification (Cls)

The final classification was performed using a Transformer-based classifier, which takes the top-K re-ranked evidence and the original claim as input and outputs one of four labels: SUPPORTS, REFUTES, NOT_ENOUGH_INFO, or DISPUTED. We used two transformer layers and added a ReLU-activated hidden layer to enhance non-linear modeling capability.

4.2 Hyperparameter and Dimension Chosen

To optimize the number of evidence passages selected for classification, we compared two configurations: Top-3 and Top-4. This was motivated by the average number of evidence per claim, which was approximately 3.3 in the training dataset. To facilitate a faster and more efficient comparison between the two Top-K settings, we initially reduced the original Word2Vec embeddings from 300 dimensions down to 128 dimensions using SVD.

As shown in Table 1, Top-4 consistently outperformed Top-3 across evidence retrieval F-score, classification accuracy, and harmonic mean. Therefore, Top-4 was chosen as the default setting for selecting relevant evidence in subsequent experiments.

Additionally, we investigated whether increasing the SVD dimension from 128 to 256 could yield further performance improvements. This modification aimed to preserve richer semantic information within the embeddings without significantly compromising training efficiency. According to Table 1, this dimensionality increase led to a modest improvement in evidence retrieval F-score (from 0.0688 to 0.0733) and harmonic mean (from 0.1192 to 0.126), whereas classification accuracy remained relatively stable.

SVD	Top-K	ER F	Cls Acc	Harm. M
128	3	0.0651	0.4351	0.1132
128	4	0.0688	0.4481	0.1192
256	4	0.0733	0.4410	0.1260

Table 1: Effect of SVD dimension and Top-K setting on evidence retrieval F (ER F), classification accuracy (Cls Acc), and harmonic mean (Harm. M) on the development set.

4.3 Attention Pooling and Regularization

4.3.1 Limitations of Mean Pooling

In the initial Transformer-based two-stage framework, we observed potential limitations of mean pooling, which aggregates token representations by treating all tokens equally. This uniform treatment might obscure important semantic information, thereby limiting the model's ability to distinguish between relevant and irrelevant evidence (Lin et al., 2017).

4.3.2 Attention Pooling Mechanism

To address this, we introduced attention pooling to replace mean pooling. This mechanism dynamically assigns weights to each token based on its semantic contribution, enhancing the model's ability to capture key connections between the claim and evidence. Specifically, attention weights are computed through a neural network with tanh activation, followed by softmax normalization.

4.3.3 Additional Hidden Layer

We further enhanced the Transformer by adding a 256-dimensional hidden layer with ReLU activation and Layer Normalization, improving model expressiveness and stability. To accommodate the added complexity, we reduced the initial learning rate from 1×10^{-4} to 5×10^{-5}.

4.3.4 Dropout Rate Effect

To assess the effect of regularization, we compared dropout rates of 0.3 and 0.4, keeping other architecture settings constant. As shown in Table 2, dropout 0.3 yielded better claim classification accuracy (0.4610 vs. 0.4416) and harmonic mean (0.1298 vs. 0.1289), with the same evidence retrieval F-score (0.0755). Higher dropout did not improve generalization but slightly degraded performance.

Dropout	ER F	Cls Acc	Harm. M
0.3	0.0755	0.4610	0.1298
0.4	0.0755	0.4416	0.1289

Table 2: Effect of Dropout Rate on ER F, Cls Acc, and Harm. M with Attention Pooling

These findings suggest that a dropout rate of 0.3 provides a better balance between regularization and model expressiveness in our architecture.

4.4 Expanding the Pre-ranking Scope: Tuning Top-K Evidence Candidates

Limitation of Initial Pre-ranking Range. In the earlier stages of our pipeline, we performed pre-ranking by computing cosine similarity between each claim and all evidence passages using SVD-reduced Word2Vec embeddings. Initially, we limited the number of retrieved candidates to the top 6–8 based on prior baselines. However, we hypothesized that this narrow range might restrict the re-ranker's ability to identify the most relevant evidence during re-ranking.

Experimental Setup for Top-K Expansion. To address this, we expanded the scope of pre-ranking and compared a wider range of candidate sizes, specifically Top-K values of 50, 100, 200, 500, 1000, 2000, and 3000. For each setting, we retrieved the top-K most semantically similar evidence passages and passed them through the Transformer-based re-ranker and classifier. We also evaluated a special configuration: Top-200 without SVD. This helped us assess the effect of removing embedding compression.

Top-K	ER F	Cls Acc	Harm. M
50	0.077	0.468	0.132
100	0.078	0.480	0.135
200	0.078	0.481	0.135
200 (no SVD)	0.073	0.442	0.125
500	0.078	0.442	0.133
1000	0.075	0.442	0.128
2000	0.077	0.442	0.130
3000	0.077	0.442	0.131

Table 3: Effect of Pre-Ranking Top-K Size and SVD Compression on ER F, Cls Acc, and Harm. M

Evaluation Results and Optimal Top-K Setting. As shown in Table 3, increasing the candidate pool from 50 to 200 improved overall performance across all metrics. The Top-200 configuration with 256-dimensional SVD-reduced Word2Vec achieved the highest Evidence Retrieval F1-score (0.078), Claim Classification Accuracy (0.481), and Harmonic Mean (0.135) on the development set. Notably, omitting SVD in the Top-200 setting led to a significant performance drop, confirming the utility of dimensionality reduction in preserving meaningful semantic information while improving efficiency.

Trade-offs and Final Selection. Performance plateaued or slightly declined when the candidate size exceeded 500. This suggests that retrieving too many candidates may introduce noise and burden the re-ranking stage with irrelevant evidence. Therefore, Top-200 with SVD-reduced Word2Vec

(256 dimensions) was selected as the optimal setting in the final model.

5 System Overview

After extensive architecture exploration and hyperparameter tuning, we finalize a two-stage pipeline for recognizing and classifying unverifiable scientific claims. This section outlines the system design, its main components, and performance evaluation.

5.1 Overall Architecture

Our system consists of two sequential modules: an evidence retrieval module and a claim classification module. Both modules are built upon Transformer backbones with attention pooling enhancements to improve semantic aggregation.

In the evidence retrieval stage, input claims are first preprocessed and encoded using pretrained Word2Vec embeddings (trained on Google News with 300 dimensions). These embeddings are reduced to 256 dimensions using SVD for efficiency. The system retrieves the top 200 candidate evidence passages based on cosine similarity, then refines the ranking using a Transformer-based reranker. The final top-4 most relevant evidence passages are selected for each claim.

In the claim classification stage, the selected evidence is concatenated with the original claim and passed to a Transformer classifier. The classifier predicts one of four labels: *SUPPORTS*, *REFUTES*, *NOT_ENOUGH_INFO*, or *DISPUTED*.

5.2 Component Details

The evidence retrieval reranker is implemented using a 3-layer Transformer with 8 attention heads. It replaces traditional average pooling with a trainable attention pooling mechanism that assigns different weights to different tokens based on their relevance. A hidden layer with Layer Normalization is added to improve non-linear expressiveness and training stability.

The classification module shares a similar architecture and outputs label probabilities using a log-softmax layer, which ensures numerical stability during multi-class prediction.

5.3 Hyperparameter Configuration

The final model uses the following hyperparameters: **SVD dimension** = 256; **Pre-ranking top-K** = 200; **Transformer layers** = 3; **Hidden dimension** = 256; **Attention heads** = 8; **Dropout** = 0.3; **Batch size** = 32 (reranker) / 16 (classifier); **Learning rate** = 5×10^{-5}; **Final Top-K** = 4.

5.4 Results On Devleopment Dataset

Table 4 summarizes the performance of different model configurations. The final model, which integrates attention pooling, Top-4 selection, and the full dataset, consistently outperforms all baselines across evidence retrieval F1, classification accuracy, and harmonic mean.

Model	ER F1	Cls Acc	Harm. Mean
TF-IDF + BiLSTM	0.0542	0.4026	0.0955
Word2Vec + BiLSTM	0.0703	0.4285	0.1208
Word2Vec + Transformer (mean pool)	0.0733	0.4410	0.1260
Word2Vec + Transformer (attn pool)	0.0755	0.4610	0.1298
Final (attn pool + Top-4 + full data)	**0.0780**	**0.4810**	**0.1350**

Table 4: Comparison of Model Configurations on ER F1, Cls Acc, and Harm. Mean (Dev Set)

5.5 Analysis

The final model benefits from several key improvements. Attention pooling outperformed mean pooling by better capturing relevant semantic information, improving **Cls Acc** by 4.5%. Expanding the pre-ranking candidate pool from 6–8 to 200 enabled the reranker to identify stronger supporting evidence. Additional hidden layers and Layer Normalization improved model stability and representation capacity. Altogether, our final model achieved a 41.3% relative improvement in **Harm. M** over the baseline. This demonstrates the effectiveness of attention-enhanced Transformer architectures for scientific claim verification.

5.6 Final Prediction On Test Set

Based on our best model, the performance on test set is: Evidence Retrieval F-score (F) = 0.0696, Claim Classification Accuracy (A) = 0.4416, Harmonic Mean of F and A = 0.1202

6 Conclusion

Overall, we designed the two-stage Transformer based system for scientific claim verification. By integrating Word2Vec embeddings, attention pooling and optimized evidence selection, our final model achieved a 41.3% relative improvement in harmonic mean over the baseline model.

7 Team Contribution

In this project, all our team members contributed evenly to all the stages of our research and report writing. All our members took part in data preprocessing, model selection and hyperparameters tuning. The work was conducted in a highly cooperative manner, ensuring equal participation and mutual support.

References

Google. 2019. Google code archive - long-term storage for google code project hosting. https://code.google.com/archive/p/word2vec/. [online] Google.com. Available at: https://code.google.com/archive/p/word2vec/.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

huggingface.co. n.d. Bert. https://huggingface.co/docs/transformers/model_doc/bert. [online] Available at: https://huggingface.co/docs/transformers/model_doc/bert.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *Preprint*, arXiv:1703.03130.

Jayant Nehra. 2024. Creating a local search engine with tf-idf and cosine similarity: A hands-on guide. https://medium.com/@jayantnehra18/creating-a-local-search-engine-with-tf-idf-and-cosine-similarity-a-hands-on-guide-21af2fba6416. [online] Medium. Available at: https://medium.com/@jayantnehra18/creating-a-local-search-engine-with-tf-idf-and-cosine-similarity-a-hands-on-guide-21af2fba6416.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. [online] Available at: https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.