SemEval-2021 Task 11: Extracting Contribution Knowledge from NLP Scholarly Articles

Group 17: Kan Yip Keng, Lin Mei An, Yang Zi Yun, Yew Kai Zhe

Abstract

In this work, we describe our attempts for the subtask 1 and 2 of SemEval 2021 Task 11: NLP Contribution Graph Challenge. Subtask 1 aims to identify the contributing sentences in a given publication using several different approaches including Naive Bayes, SciBERT, Sentence-BERT and BiLSTM. Subtask 2 extracts the scientific terms and predicate phrases from the contributing sentences using models such as BERT, SciBERT, BiLSTM and CRF with BIO sequence labelling scheme.

1 Introduction

The Knowledge Graph (KG) describes the concepts, entities and their relationships in raw data. As the rate of research publications grows exponentially, it can be useful to represent knowledge efficiently with KGs. The goal of the NLPContributionGraph (NCG) task is to develop automated systems that structure contributions from NLP scholarly articles in the English language. Our project consists of 2 subtasks: Sentences Extraction (subtask 1) and Phrases Extraction (subtask 2).

2 Background

2.1 Problem Definition

Consider a document $\mathrm{D} = \{\mathrm{s}_1, \mathrm{s}_2, .., \mathrm{s}_\mathrm{i}, .., \mathrm{s}_\mathrm{N}\}$ having N sentences s_i . Subtask 1 finds M contribution sentences denoted by $\mathrm{S} = \{\mathrm{s}_1, \mathrm{s}_2, .., \mathrm{s}_\mathrm{i}, .., \mathrm{s}_\mathrm{M}\}$ from D. Subtask 2 selects phrases $\mathrm{P} = \{\mathrm{p}_1, p_2, .., p_\mathrm{i}, .., p_L\}$ where p_i is a phrase selected from a sentence $\mathrm{s} \in \mathrm{S}$ and L is total number of phrases in the sentence s. We approached subtask 1 as an extractive summarization problem, which can also be viewed as a binary classification task to label contributing and non-contributing sentences in the scientific text as 1 and 0 respectively. For Subtask 2, we tackled the problem as a sequence labelling problem, such as named-entity recognition task. We tag each word in the input sentence with the BIO labelling scheme.

2.2 Related Work

We explored supervised learning approaches for text classification, which can be divided into parametric and non-parametric approaches. Parametric models like logistic regression, naive Bayes assume an underlying function/distribution. They are simpler, more inflexible than parametric methods but are also more interpretable. In contrast, non-parametric models like k-nearest neighbors and neural networks do not assume the regression function and are more flexible but less interpretable and have a risk of overfitting the data. We use a parametric model (naïve Bayes) as the baseline model and try to improve performance using non-parametric models (neural networks).

Pre-trained language models have shown to be useful in learning common language representations by utilizing a large amount of unlabeled data. BERT is based on a multi-layer bidirectional Transformer (Vaswani et al., 2017) and is trained on plain text for masked word prediction and next sentence prediction tasks.

We also explored the current state-of-the-art for named entity recognition tasks, which is a combination of SciBERT, BiLSTM and CRF (Xu L., et al., 2021). Each word token in the input sentence is embedded with SciBERT and passed into the BiLSTM model for feature extraction. The feature vector is then fed into a CRF model to predict the BIO tags for the word tokens.

3 System Overview

3.1 Data Exploration

The official NCG shared task provided a dataset of 237 NLP scholarly articles. Each article has a set of contributing sentence ids and each contributing sentence has a set of phrases.

Token Length	% of sentences less than token length
50	94.57
100	99.74
150	99.93
200	99.96

Token Length	% of documents less than token length
6000	84.16
7000	95.25
8000	97.96
9000	99.55

Table 1: Token length statistics

Statistics	
Documents	237
Contribution sentences	5096
Non-contribution sentences	50105
Avg. sentences in document	232.915
Avg. tokens in sentence	20.622
Avg. contribution sentences in document	21.38
Avg. phrases in document	128.53
Avg. IU in document	4.43
Max tokens in sentence	396

Table 2: Train dataset statistics

We can see that most sentence lengths are below 100 tokens. This informs the maximum token length of the BERT models chosen for both subtasks. Furthermore, there is an extreme class imbalance of 1:10 of contribution to non-contribution sentences. This means that strategies for correcting class imbalances are important for subtask 1. The dataset is also relatively small, meaning that it would be recommended to use pretrained models for transfer learning for better generalization.

3.2 Data Preprocessing

3.2.1 Rebalancing

For subtask 1, the data provided suffers from class imbalance because most of sentences in the article are not contributing sentences. We resample the data with replacement such that contributing and non-contributing sentences have equal occurrences in the resampled dataset.

On the other hand, the inconsistencies in the human annotated phrases dataset for subtask 2 requires a more general approach in predicting scientific phrases.

3.3 Optimizers

3.3.1 Adam

We tried using SGD for finetuning, but it was too slow to converge under practical conditions and yielded poorer results. Adaptive gradient methods like Adam tend to have faster convergence than SGD, but usually have poorer generalization ability empirically (Wilson et al., 2017).

3.3.2 AdamW

Wilson et al. (2017) suggested that adaptive gradient methods do not generalize as well as SGD with momentum when tested on a diverse set of deep learning tasks. Loshchilov et al. (2019) show that a major factor of the poor generalization of Adam is since L2 regularization is not nearly as effective for it as for SGD. AdamW uses decoupled weight decay instead of L2 regularization and generalizes better empirically.

3.4 Subtask 1: Extraction of Contributing Sentences

3.4.1 Naïve Bayes

We used a naïve Bayes model, which takes in TF-IDF word embeddings (features=5000), as a simple baseline classifier. TF-IDF (term frequency–inverse document frequency) is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The TF-IDF value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word. We also do basic data preprocessing by converting sentences to lowercase and removing stop words.

3.4.2 SciBERT

SciBERT is a BERT based model trained on a large corpus of scientific texts. SciBERT has been trained on 1.14M scientific papers from Semantic Scholar corpus, which has 18% papers from the computer science domain. Hence, the classification task can benefit from the domain-specific pretraining data. However, we face a drawback due to the token limit (512) since max document length in the train dataset is ~10,000. We use SciBERT with a regression head (a linear layer on top of the pooled output). Each input consists of a sentence-label pair. The sentence does not have additional contextual information such as titles or headers. The labels are 1, 0 for contributing and non-contributing sentences respectively. We used a token limit of 100 based on exploratory data analysis, where 99.7% of sentences are within 100 tokens.

3.4.3 Sentence-BERT (SBERT)

We considered Long Document Transformers such as Reformers and Longformers to encode the entire document as context. However, these models only support token lengths up to 4,096 in contrast to the maximum document length of ~10,000. We in turn considered sentence-level embeddings to encode the entire document.

Sentence-BERT is a Siamese BERT-network architecture, containing two identical BERT subnetworks with the same parameters and weights. Parameter updating is mirrored across both sub-networks. The produced sentence embeddings are semantically meaningful and can be compared with cosine-similarity.

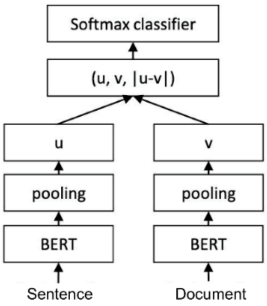

Figure 1: Architecture for Sentence-BERT classifier model

We apply mean pooling separately for sentence and document embeddings u, v respectively from SBERT, then concatenate embeddings as such: (u, v, |u-v|). The motivation is that we want the model to learn a shared distance metric embedding space for both documents and sentences such that embedding for documents and contribution sentences in that document are close in distance. These embeddings can then be passed into a classifier.

One drawback is that this model is not pre-trained on a domain-specific dataset unlike SciBERT. We use a pretrained Sentence-BERT model all-mpnet-base-v2 based on the microsoft/mpnet-base model, with additional fine-tuning on a large and diverse dataset of over 1 billion training pairs.

We experimented with using gsarti/scibert-nli, which is SciBERT fine-tuned on the SNLI and the MultiNLI datasets to produce universal sentence embeddings. The SNLI is a collection of 570,000 sentence pairs annotated with the label's contradiction, entailment, and neutral. MultiNLI contains 430,000 sentence pairs and covers a range of genres of spoken and written text. However, it yielded poorer results than the all-mpnet-base-v2 model, perhaps because scibert-nli was trained ~4 hours on the NVIDIA Tesla P100 GPU, while all-mpnet-base-v2 was trained on a TPU v3-8 for 100k steps using a batch size of 1024 (128 per TPU core) on a much larger dataset.

3.4.4 SciBERT + BiLSTM

An extension of SciBERT, this approach was used by one of the competition teams (Shailabh, et al., 2021). Stacking the BiLSTM model on top of the SciBERT model helps to encode hidden semantics and long-distance dependencies. Sentences are processed using SciBERT and passed through stacked BiLSTM layers. A dropout layer is applied to avoid overfitting before further processing through linear layers with ReLU. The final linear layer outputs a sequence of length 2 each corresponding to the score of the labels 0 and 1.

3.5 Subtask 2: Extraction of Phrases

3.5.1 BIO sequence labelling scheme

For each golden contribution sentences, we label their ground truth phrases (the scientific terms and relational predicates) by using BIO (B=start token of phrase, I=continuation tokens of phrase and O=non-phrase tokens) sequence labelling scheme.

3.5.2 NER task state-of-the-art

We take inspiration from state-of-the-art for named entity recognition task proposed by Lei Xu, et al (2021) which uses a combination of BERT, BiLSTM and CRF (Lafferty et al., 2001). We also refer to the implementation by the IITK team (Shailabh, et al., 2021).

First, the input sentence is tokenized by BERT tokenizer, which is a WordPiece tokenizer which is robust to Out-of-Vocabulary (OOV) tokens shown by Schuster and Kaisuke (2012). The first sub-token for each word is passed into the BERT model, which gives an embedding of size 768 for each token. The embeddings are then fed into the

BiLSTM layer for feature extraction, which reduces the dimension to 200. The output of BiLSTM hidden layer is transformed by an extra hidden layer into a 5-dimension layer as the emission probability of the 5 tags, namely the start, end, B, I and O tags. Lastly, the CRF (Conditional Random Field) layer predicts the input word's tag by using the the emission provided by the BiLSTM and transition probability from a 5 x 5 matrix with trainable parameters which is trained jointly. We use the probability distribution of each tag and the one-hot vector of ground truth tags to calculate the negative log likelihood for the loss of each word prediction.

Additionally, we implemented the Viterbi algorithm to improve the efficiency of predicting the tags during forward propagation.

To study the importance of each component in our model architecture, we compare the performance of using BERT vs SciBERT to generate token embedding. Since BERT is a bidirectional model by itself, we are interested to find out the whether the BiLSTM layer is necessary. Hence, we compare the model performance with or without the BiLSTM layer.

4 Experimental Setup

4.1 Models and Parameters

For hyperparameters for finetuning BERTs, we use a lower learning rate, such as 2e-5, to make BERT overcome the catastrophic forgetting problem (Chi Sun, et al., 2019). Catastrophic forgetting is a common problem in transfer learning where pre-trained knowledge is erased during learning of new knowledge. For specific hyperparameters for the respective BERT variants, we refer to the original papers. Specific details on the training of the models can be found in the appendix.

4.2 Evaluation Metrics

Given the imbalanced dataset, accuracy is not a good metric of the model's usefulness (91% accuracy with all 0 labels for subtask 1), hence we use the F1-score, which is the harmonic mean of precision and recall.

5 Results

Approach	**F1-Score**
Naïve Bayes	0.326
SciBERT	**0.437**
SentBERT	0.368
SciBERT + BiLSTM	0.422

Table 3: Results for Subtask 1

Approach	**F1-score**
BERT + CRF	0.730
BERT + BiLSTM + CRF	0.734
SciBERT + CRF	0.748
SciBERT + BiLSTM + CRF	**0.751**

Table 4: Results for Subtask 2

6 Conclusion

For subtask 1, future work could explore other methods of data augmentation for correcting class imbalance, such as synonym replacement to replace words in the input sentences with common synonyms. Other ways to account for class imbalance include weighted loss functions and adversarial training which can improve generalization (Miyato et al., 2017).

One surprising finding is that the SciBERT model with a regression head performed the best on the validation data, even better than the SciBERT + BiLSTM model used in the competition. This might be dependent on the train/validation split, especially since the dataset is small. To better estimate test set error, k-fold cross validation can be used.

For subtask 2, SciBERT consistently outperforms BERT because it is pre-trained on scientific publications which fits better into our problem context as compared to the more generalized BERT. Besides that, BiLSTM also works well in tandem with BERT/SciBERT models by providing a slight boost to the F1-score.

In future work, improvements can be done by ensembling multiple models for majority voting to enhance the accuracy of model by reducing variance and bias while maintaining its generalization.

7 Appendix

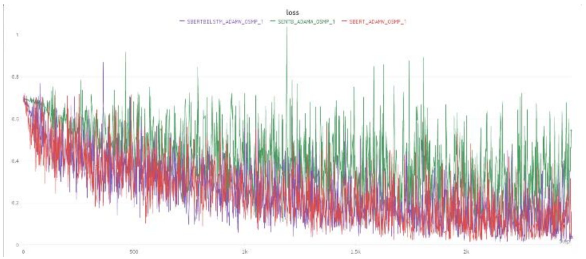

Figure 2: Subtask 1 training loss graph

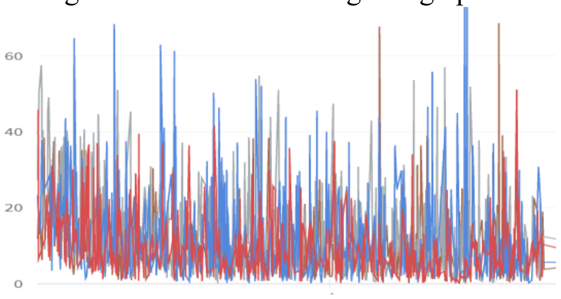

Figure 3: Subtask 2 training loss graph

7.1 Training details

7.1.1 Subtask 1

7.1.1.1 SciBERT

We trained SciBERT for 2 epochs, with a learning rate of 2e-5 and batch size of 32 using the AdamW optimizer. Class imbalance of training data was corrected with oversampling. The loss function is cross-entropy loss.

7.1.1.2 Sentence-BERT

We trained Sentence-BERT for 1 epoch, with a learning rate of 2e-5 and batch size of 16 using the AdamW optimizer. Class imbalance of training data was corrected with oversampling. The loss function is binary cross-entropy loss.

7.1.1.3 SciBERT+BiLSTM

We trained SciBERT+BiLSTM for 2 epochs, with a learning rate of 2e-5 and batch size of 32 using the AdamW optimizer. Class imbalance of training data was corrected with oversampling. The loss function is cross-entropy loss.

7.1.2 Subtask 2

7.1.2.1 BERT/SciBERT+BiLSTM+CRF

We trained BERT/SciBERT+BiLSTM+CRF for 3 epochs, with a learning rate of 2e-5 using the AdamW optimizer. BERT/SciBERT embedding dimension is 768. BiLSTM hidden dimension is 200.

7.1.2.1 BERT/SciBERT+CRF

We trained BERT/SciBERT+BiLSTM+CRF for 3 epochs, with a learning rate of 2e-5 using the AdamW optimizer. BERT/SciBERT embedding dimension is 768. We added a hidden feed forward neural network with 200 dimensions to replace the BiLSTM layer mentioned in 7.1.2.1.

References

1. Beltagy, Cohan, &amp; Lo. (n.d.). SCIBERT: A Pretrained Language Model for Scientific Text. Retrieved November 14, 2021, from https://arxiv.org/pdf/1903.10676v3.pdf.

2. How to fine-tune bert for text classification? - arxiv.org. (n.d.). Retrieved November 14, 2021, from https://arxiv.org/pdf/1905.05583.pdf.

3. John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

4. Loshchilov, I., &amp; Hutter, F. (2019, January 4). Decoupled weight decay regularization. arXiv.org. Retrieved November 14, 2021, from https://arxiv.org/abs/1711.05101v3.

5. Miyato, T., Dai, A. M., &amp; Goodfellow, I. (2017, May 6). Adversarial training methods for semi-supervised text classification. arXiv.org. Retrieved November 14, 2021, from https://arxiv.org/abs/1605.07725.

6. Reimers, &amp; Gurevych. (n.d.). Sentence-bert: Sentence embeddings using ... - arxiv.org. Retrieved November 14, 2021, from https://arxiv.org/pdf/1908.10084.pdf.

7. Shailabh, Modi, &amp; Chaurasia. (n.d.). ArXiv:2104.01619v1 [cs.CL] 4 Apr 2021. KnowGraph@IITK at SemEval-2021 Task 11: Building Knowledge Graph for NLP Research. Retrieved November 14, 2021, from https://arxiv.org/pdf/2104.01619.pdf.

8. Vaswani, Polosukhin, Kaiser, Gomez, Jones, Uszkoreit, Parmar, &amp; Shazeer. (n.d.). Attention is all you need - nips. Retrieved November 14, 2021, from https://papers.nips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

9. Wilson, Recht, Srebro, Stern, &amp; Roelofs. (n.d.). The marginal value of adaptive gradient methods ... - neurips. Retrieved November 14, 2021, from https://proceedings.neurips.cc/paper/2017/file/81b3833e2504647f9d794f7d7b9bf341-Paper.pdf.

10. Schuster and Kaisuke. Japanese and Korean Voice Search. Retrieved 2012, from https://static.googleusercontent.com/media/research.google.com/ja//pubs/archive/37842.pdf