LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Fine-Tuned Sentence Transformer Model for Question Answering Task

Ercong Nie

e.nie@campus.lmu.de

Master Student

Center for Information and Language Processing, Faculty of Languages and Literatures, LMU of Munich

May 26, 2022

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Contents

Introduction

Language Models (LMs)

- **LM** is a core topic in the computational linguistic / natural language processing (NLP) field.
- By applying different types of LMs, many various NLP tasks can be addressed (at least to some extent).

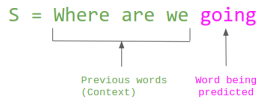
Definition

Language Models are models that assign possibilities to sequences of words[1].

Basically, LMs are used for **prediction**.

[1] Jurafsky and Martin (2008)

Language Models



$$P(S) = P(\text{Where}) \times P(\text{are} \mid \text{Where}) \times P(\text{we} \mid \text{Where are}) \times P(\text{going} \mid \text{Where are we}) \quad 2$$

- Predicting the probability of a sentence.
- Predicting the probability of a word given its context by applying the **chain rule of probability**.
- Predicting the probability of a word given its previous word by applying the **Markov assumption**, i.e. the **bigram model**.

[2] https://humanloop.com/blog/how-good-is-gpt-3-in-practice

Different Types of Language Models

- N-Gram Language Models
  - **n-gram**: a sequence of n words
  - Predicting the probability of one word based on its previous $(N-1)$ words.
- Neural Network Based Language Models
- Transformer-Based Language Models

Different Types of Language Models

- **N-Gram Language Models**
- **Neural Network Based Language Models**
  - Predicting the next word in a sequence by its previous words based on neural network structures, like **RNN**, **LSTM**...
  - Adopting a more powerful method to represent words, i.e. **the word embedding**, a semantically meaningful vector.
- **Transformer-Based Language Models**

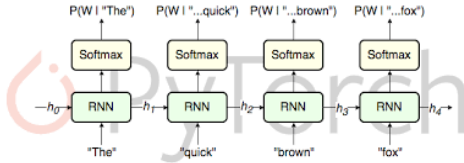


Figure 1: An Example of RNN language model structure[a]

[a] https://medium.com/@florijan.stamenkovic_99541/

Different Types of Language Models

- **N-Gram Language Models**
- **Neural Network Based Language Models**
- **Transformer-Based Language Models**
  - **Transformer**: A network structure based on the **self-attention** mechanism.
  - An improvement compared with NN-based models: A sequence processing method that eliminates recurrent connections.

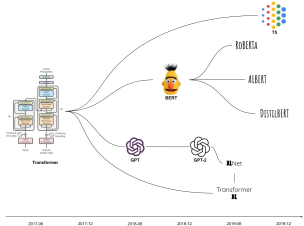


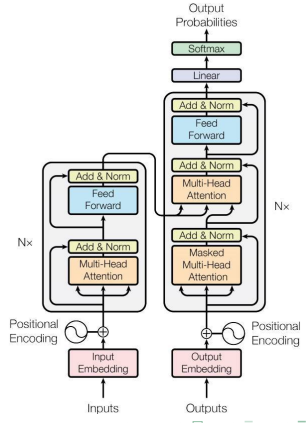
Figure 2: Collection of some transformer-based models[a]

[a] https://www.factored.ai/2021/09/21/

Transformer-Based Language Models

Structure of Transformer Model[3]

- Composed of two parts, **Encoder** (left) and **Decoder** (right).
- Both parts contain modules that can be stacked on top of each other multiple times.
- Core module parts: Multi-head attention and feed forward layers.
- Inputs and Outputs first embedded into an n-dim space.
- Positional embedding offers positional information.



Structure of BERT

- BERT: Bidirectional Encoder Representations from Transformers
- Has two pre-training tasks:
  1. **Masked Language Modeling (MLM)**
  2. **Next Sentence Prediction (NSP)**
- An autoencoder model, self-supervised training objectives (no need for labeled data in pre-training)

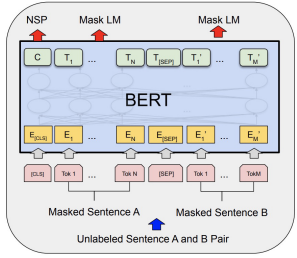


Figure 3: The structure of BERT[ab]

[a] https://paperswithcode.com/lib/allennlp
[b] Devlin et al. (2016)

Sizes of Transformer-Based Models

Year	Model	# of Parameters	Dataset Size
2019	BERT [39]	3.4E+08	16GB
2019	DistilBERT [113]	6.60E+07	16GB
2019	ALBERT [70]	2.23E+08	16GB
2019	XLNet (Large) [150]	3.40E+08	126GB
2020	ERNIE-GEN (Large) [145]	3.40E+08	16GB
2019	RoBERTa (Large) [74]	3.55E+08	161GB
2019	MegatronLM [122]	8.30E+09	174GB
2020	T5-11B [107]	1.10E+10	745GB
2020	T-NLG [112]	1.70E+10	174GB
2020	GPT-3 [25]	1.75E+11	570GB
2020	GShard [73]	6.00E+11	–
2021	Switch-C [43]	1.57E+12	745GB

Figure 4: Overview of sizes of recent large language models[4]

[4] Bender et al. (2021)

Pre-Training Fine-Tuning Paradigm

- First **pre-training** model on large **general raw** language data.
- Then **fine-tuning** the model's parameters by using relatively small amount of **labeled** data so that the model can be applied in a **specific** field or task.
- Basically a **transfer learning** method.
- Most transform-based models adopt this paradigm, so they are also called **Pretrained Language Models (PLMs)**.

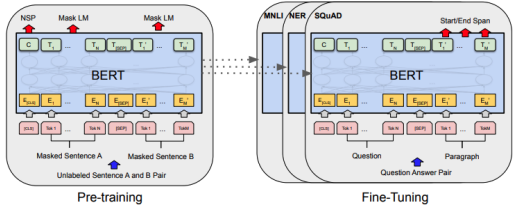


Figure 5: Pre-training Fine-tuning Paradigm Devlin et al. (2016)

Sentence Transformers

Motivation

- Original transformer-based models can only encode texts on the **word** level.
- Many language processing tasks are handled on the **sentence** or even document level, such as question answering, document classification, sentiment analysis etc.

Overview

- Modification of pre-trained language models.
- Use siamese and triplet network structures
- Derive semantically meaningful **sentence** embeddings that can be compared to each other by using cosine-similarity.

Structure of SBERT

- Add a **pooling** operation to the output of BERT.
- Derive a **fixed** size sentence embedding.
- Create siamese and triplet networks to update the weights in the fine-tuning.

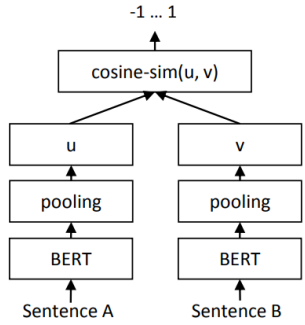


Figure 6: SBERT Architecture. Reimers and Gurevych (2019)

An Application of Sentence Transformers in Question Answering Task

Task Description

Question Answering (QA)

- Basically a system that allows a user to ask a question in natural language and receive an answer quickly[5].
- A classical NLP task that can be applied in many scenarios, such as search engine, chatbot, information retrieval, conversational AI etc.
- Another perspective: answer matching and selection[6].

[5] Hirschman and Gaizauskas (2001)
[6] Feng et al. (2015)

Task Description

Goal

- Fine-tune a pre-trained sentence transformer model that can be used for **selecting the proper answers** to a given question **from a given pool of candidate answers**.

Subtasks

1. How to select and fine-tune a sentence transformer?
2. How is the fine-tuned sentence transformer model used for answer selection?

Dataset Introduction I

- **Dataset**: **InsuranceQA**, created by IBM Watson researchers.
- **Contents**: Questions and answers in the **insurance** domain, original data from the website Insurance Library.
- Composed of training and evaluation parts. Each question has a **pool** of **100** possible answers, some of which are true answers (ground truth).

Dataset Introduction II

- **Data Structure**: <Domain><TAB><QUESTION><TAB><Groundtruth><TAB><Pool> <QUESTION> is represented by the tokens in the form of **index**. <Groundtruth> and <POOL> are represented by the **labels** of answer.

```
renters-insurance   idx_1285 idx_1010 idx_999 idx_136 idx_65807 22542 4380  2235 26739 24916 17855 3406 21201 70 19553 22008
5768 18118 2105 20821 2316 25072 6805 9846 24262 6317 6250 13690 1467 4770 10917 18784 7229 8030 26792 15729 13179 2464 3884
23403 24493 6292 24533 26 13983 15294 26821 22449 20057 10641 16504 8153 14453 6276 16349 27141 14698 13650 12175 836 17050
15911 5230 10139 5955 18903 2710 471 20269 11888 17073 16128 8026 26441 19693 11405 7745 14596 13353 22175 5530 17982 10727
18225 12703 8782 26875 16985 1324 16967 25308 7420 17279 11137 26769 22418 18674 21014 14694 20737 22135 3346 7342 5099 12388
20032 12029
```

Figure 7: Example of a data item

- 12,889 questions in training set, 4,000 questions in evaluation set, 27,413 answers totally.

Data Preparation for Fine-Tuning

- Convert the original training set into the form of sentence pairs with a label.
- For each question in the original training set, create two sentence pairs as the inputs for model fine-tuning.
  1. Select the first answer from the <Groundtruth> to compose the **positive** sentence pair (labeling 1).
  2. Randomly select one answer from the <POOL> to compose the **negative** pair (labeling 2).

Model Selection

- First select a pre-trained sentence transformer for later fine-tuning.
- Research on the zero-shot performance of different transformers on the insurance QA task.
- **Accuracy** and **running time** are considered.

No.	Model Name	Accuracy	Running Time
1	average_word_embeddings_komninos	0.0875	**58.271 s**
2	paraphrase-distilroberta-base-v2	0.2417	4440.939 s
3	paraphrase-MiniLM-L3-v2	0.2241	1182.681 s
4	paraphrase-MiniLM-L6-v2	0.2447	1827.053 s
5	paraphrase-MiniLM-L12-v2	0.2661	3117.037 s
6	paraphrase-mpnet-base-v2	**0.2825**	9489.678 s
7	paraphrase-TinyBERT-L6-v2	0.2664	5069.734 s
8	stsb-mpnet-base-v2	0.1575	7183.301 s
9	stsb-roberta-base-v2	0.1507	6456.512 s

Table 1: Performance of different pre-trained Sentence Transformer models on the insurance QA task.

Fine-Tuning Details

- **Model**: Fine-tune the pre-trained sentence transformer "paraphrase-TinyBERT-L6-v2"
- **Training data size**: 25,778 question-answer pairs
- Batch size: 64
- Epoch: 1
- Loss function: cosine similarity

Evaluation and Result I

Evaluation Method

- Use the fine-tuned sentence transformer to **encode** the question and all answers in the pool.
- Compute the cosine similarity score between all answers and the given question and select the answer with the highest score as the predicted answer.
- Check if the predicted answer is in the ground truth set.

Evaluation and Result II

Pre-trained Model without Fine-Tuning	
Name	paraphrase-TinyBERT-L6-v2
Accuracy.	0.2664
Runtime	5069.734 s
Pre-trained Model with Fine-Tuning	
Name	paraphrase-TinyBERT-L6-v2
Accuracy.	0.4867
Runtime	5651.248 s

Table 2: Result of fine-tuned Sentence Transformed applied to the insurance QA task.

Result By fine-tuning with the domain data, the accuracy of the QA system has been significantly improved, rising from **0.2664** to **0.4867** with an increasing of 82.7%.

Summary

Summary

- Language models are basically used for predicting the probability of texts or words and are widely applied in the NLP field.
- Transformer-based language models are most commonly used for various NLP problems nowadays, especially in the "Pre-training fine-tuning paradigm".
- Sentence Transformer is a modification of pre-trained language models to derive semantically meaningful sentence embeddings.
- The experiment of the fine-tuned sentence transformer applied on InsuranceQA corpus proves that through fine-tuning, the sentence transformer has learned stronger representation ability in the insurance domain.

References I

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2016). Bert: Bidirectional encoder representations from transformers.

Feng, M., Xiang, B., Glass, M. R., Wang, L., and Zhou, B. (2015). Applying deep learning to answer selection: A study and an open task. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 813–820. IEEE.

References II

Hirschman, L. and Gaizauskas, R. (2001). Natural language question answering: the view from here. *natural language engineering*, 7(4):275–300.

Jurafsky, D. and Martin, J. H. (2008). Speech and language processing: An introduction to speech recognition, computational linguistics and natural language processing. *Upper Saddle River, NJ: Prentice Hall*.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

References III

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Thanks for your attention!