BERT Tech Review

Author: Daocheng Wang

Introduction

BERT, stands for Bidirectional Encoder Representations from Transformers, is a language model introduced by Google in 2019. I would like to do an overview about the background of BERT, how BERT works, and its functionalities. Then I will summarize the existing approaches of generating domain specific models via BERT, and finally I will propose a way to approximate a collection language model containing the mappings between words and probabilities using BERT.

Background of BERT

In terms of the background of BERT, pre-training language models has been an effective way to improve NLP tasks. To apply pre-trained language representations to downstream tasks, certain existing language models, including OpenAI GPT, use fine-tuning, which introduces task-specific parameters and fine-tune all pre-trained parameters during downstream tasks; however, the effectiveness of the existing language models has been limited due to being unidirectional, which limits the choice of architecture for pre-training. To address this limitation, BERT is introduced, which utilizes a ‘masked language model’ (MLM) pre-training objective to pre-train a deep bidirectional transformer, which takes both pre-context and post-context of a sentence into account.

How BERT works

Regarding how BERT works, let’s first look into MLM. MLM is achieved via hiding certain tokens from the input randomly, and the objective is to predict and recover the original vocabulary of the hidden tokens based on the context. Upon completing the pre-training of a deep bidirectional transformer, another task ‘next sentence prediction’ (NSP) is adopted to allow understanding of sentence relationships and jointly pre-train text-pair representations. After the pre-training step of using MLM for BERT, the next procedure is the fine-tuning step. Since the bidirectional transformer obtained from the pre-train step allows BERT to model many downstream tasks, fine-tuning becomes relatively easy and inexpensive. During the fine-tuning step, BERT is first initialized with the pre-trained parameters, which will be fine-tuned via the labeled data from the downstream tasks. Because of the pre-training and NSP steps, BERT was able to perform successfully on a number of natural language processing tasks.

BERT Use Cases

I would like to summarize the downstream tasks that BERT can effectively solve, including MLM, question answering, and multiple choice questions. Given that BERT was pre-trained via MLM, BERT has been very effective in recovering the hidden vocabularies from masked sentence(s). In addition, BERT performs question answering tasks very well. Upon receiving the input of a question and a passage containing the answer, BERT is able to predict the answer text span in the passage with state-of-the-art performance, as demonstrated using SQuAD (Stanford Question Answering Dataset). Furthermore, besides question answering, BERT can also solve multiple choice questions. Based on the results obtained from Situations With Adversarial Generations (SWAG) dataset, BERT can predict the most plausible continuation of a sentence among four choices with higher accuracy than existing language models including ESIM+ELMo and OpenAPI GPT.

Extending BERT to domain-specific models

One of the limitations of BERT is that the standard text corpus used to train BERT might not be sufficient for the use cases of domain-specific text needs. To address this limitation, a number of efforts have been made. Some researchers have trained domain-specific models. These models are generated by training BERT architecture from scratch on a domain-specific corpus instead of the original standard text corpus. Some examples include SciBERT for biomedical and computer science literature, FinBERT for financial services, BioBERT for biomedical literature, and so on. Apart from the existing trained domain-specific models, a training method called exBERT [2] is also introduced aiming to extend BERT pre-trained models to a new pre-trained model for a specific domain with new vocabularies using a small extension module, which can learn to adapt an augmenting embedding for the new domain.

Proposal on approximating a collection LM via BERT

I am proposing a way to approximate a collection LM via BERT. As shown from the above 2 approaches on extending BERT, in order to generate domain-specific models, certain training is needed regardless of a full training from scratch or partial training with the help of a module. In addition, given that BERT is a generative LM with the need of bidirectional context, there's no easy way to directly retrieve the word distribution (the mapping between vocabulary and its probability) without context. Thus I am proposing a way to approximate a collection LM without explicit training. Upon getting a fairly large collection of corpus, we can mask 15% of the tokens in the corpus and then leverage the MLM feature of BERT to recover those 15% hidden tokens. For each masked token during the MLM process, BERT will produce a list of possible vocabularies with corresponding probabilities, which can be used to aggregate and then approximate the collection LM. In this way, no explicit training is needed; instead, the collection LM can be generated by only utilizing the MLM functionality of BERT.

Conclusion

In conclusion, this tech review has covered the overview of BERT, the state-of-the-art approaches to use or generate domain-specific BERT models as well as a proposal on approximating a collection LM by leveraging the MLM feature of BERT instead of training explicitly.

References

1. Devlin, Jacob; Chang, Ming-Wei; Lee, Kenton; Toutanova, Kristina (11 October 2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". arXiv:1810.04805v2 [cs.CL].
2. Tai, Wen; Kung, H. T.; Dong, Xin; Comiter, Marcus; Kuo, Chang-Fu (November 2020). "exBERT: Extending Pre-trained Models with Domain-specific Vocabulary Under Constrained Training Resources". *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics: 1433–1439. doi:10.18653/v1/2020.findings-emnlp.129. S2CID 222305413.
3. "Domain-Specific BERT Models · Chris McCormick". *mccormickml.com*. Retrieved 2022-11-05.