A Review on BERT and Its Implementation in Various NLP Tasks

Vrishali Chakkarwar[1](✉), Sharvari Tamane[2], and Ankita Thombre[1]

[1] Department of Computer Science and Engineering, Government College of Engineering, Aurangabad, Maharashtra, India
vrush.a143@gmail.com, thombreankita29@gmail.com

[2] University Department of Information and Communication Technology MGM University, Aurangabad, Maharashtra, India
sharvaree73@yahoo.com

Abstract. We present a detailed study of BERT, which stands for 'Bidirectional Encoder Representations from Transformers'. Natural Language Processing can be considered as an important field when concerned with development of intelligent systems. Various tasks require understanding the correct meaning of the sentence in order to provide the output. Languages are difficult to understand by computers due to their ever-changing nature with context. BERT is considered as a revolution for making the computers understand the context of the text, which is the biggest hurdle in natural language processing tasks. It learns the language and its context in a way that closely resembles how a human brain understand the meaning of a sentence. It is unique because of its ability to learn from both left and right context in a sentence for a specific word. The evolution of BERT has marked a new era in perception and understanding of natural languages which may lead computers to grasps the natural language with better comprehension. The purpose of this paper is in the effort of providing a better understanding of BERT language model and its implementation in various NLP tasks.

Keywords: Bidirectional Encoder Representations from Transformers (BERT) · Transformers · natural language processing (NLP) · text summarization · text classification · sentence similarity · distillBERT

1 Introduction

The credit for making computer systems capable of comprehending the natural text, its semantics and context goes to Natural Language Processing. The field of NLP has seen various developments in the past decades and still has a big scope for improvement. Various techniques like use of recurrent neural networks, long short-term memory, sequence 2 sequence (Seq2Seq) architecture, transformer based Generative Pre-Trained Transformer (GPT Open AI), Embeddings from Language Model (ELMo) and so on, have been introduced in past decades.

Transformers marked a major breakthrough in NLP tasks. Transformers are a neural network-based architecture, with two major components namely- Encoder and Decoder.

S. Tamane et al. (Eds.): ICAMIDA 2022, ACSR 105, pp. 112–121, 2023.
https://doi.org/10.2991/978-94-6463-136-4_12

The Transformers used as Language models in GPT, generally consist of a decoder stack, meaning numerous decoders are stacked one above another, each one taking the output of previous one as an input. They implement feed forward neural networks at each encoder and decoder along with self-attention. This combination enables the system to extract meaningful context of words, on a sentence level.

The above-mentioned techniques gave good results, but were unidirectional in nature. Meaning, these models read either left to right or right to left. Considering the altering nature of languages, this was not sufficient. For example, if we consider a sentence to be fed to Transformer Language Model (LM), it will compute a vector and assign a meaning to the word based on what is has already read so far. In such scenario, if the real context of sentence is based on the words or token yet to be read, the model will extract an incorrect meaning of the sentence. Coming to human mind, it looks forward as well as backwards when deriving the meaning of a text. Thus, when applying this to machines, it is important to take into consideration the context from both the directions, so as to assign the token a correct context or a correct form in sentence level, word prediction task for an instance.

We take a simple example to clarify the need of bidirectional model. Consider, two sentences –

1) The boy sat on the left side.
2) She left her bag on the chair.

When these two sentences are given as an input to simple LM, the context assign to the word "left" in both the cases is in the sense of direction. In case of unidirectional LM working left to right, this context is fixed based on the sub string - 'boy sat on' in statement one and 'She' in statement two. Whereas, in contextual LM like working left to right, the meaning of the word 'left' determines the context of the rest of the sentence, which changes the entire statement 2 as- the bag was on left of the chair.

Such ambiguity can be harmful if applied as it is in higher level NLP tasks. Thus, to resolve such ambiguities, a bi-directional Language Model with ability of retaining information was proposed as BERT.

2 BERT

BERT or Bi-directional Encoder Representations from Transformers was proposed by Jacob Devlin [1] in 2018.It is the first deeply bidirectional, unsupervised NLP model that makes use of transfer learning method. As stated in the original paper, 'BERT obtains a state-of-art results in 11 NLP tasks.'

BERT is a pre-trained, bi-directional language model, capable of giving consideration to the context of tokens that appear before and after a selected token. In simple words we can say, BERT can read a text in left to right and right to left direction, giving each word a proper meaning by keeping in mind both contexts, in one single iteration. In addition to being deeply bidirectional, it also has a unique feature that it can be easily applied to perform any new NLP task with slightest changes to the pre-training model. These slightest changes often consist of applying an additional output layer to the pre-trained model.

The architecture of BERT comprises of numerous encoders stacked on one another, in a feed-forward fashion, the output of one encoder is fed as input to the next. The results were analyzed on two model sizes – $\mathrm{BERT}_{\mathrm{BASE}}$ with 12 layers of transformer blocks, 768 hidden units and 12 attention heads; and $\mathrm{BERT}_{\mathrm{LARGE}}$ with 24 encoder blocks, 1024 hidden units and 16 attention heads. For implementation of BERT, the paper proposes two frameworks, namely– pre-training and fine-tuning.

Data used for training was composed of plain Wikipedia text corpus and Google's BookCorpus. While pre-training BERT, the training was mainly focused on two unsupervised NLP task – Masked Language Model (MLM) and Next Sentence Prediction. The deep bidirectional method forces a word to indirectly "see-itself". Thus, it randomly masks a few words (WordPiece tokens) in the sentence and tries to predict the word that should replace the masked word. This is known as the Masked Language Model and BERT has a probability of each word or token in an input sequence to be masked as 15%. Besides, to avoid any mismatch between the two frameworks, it replaces the token to be masked with a [MASK] 80% of times and by a random token rest of the time. This helps the model to comprehend the meaning of a word when it appears in more than one context, thus making it capable of keeping up with ever-changing nature of languages.

The next task considered while pre training was next sentence prediction (NSP) where given two sentences as input, the model was to output whether the second sentence follows the first one or not. The purpose was to capture the relationship between the input text. For this correct understanding of the context of individual sentences is important and can be achieved due to deep bi-directional feature. Taking this one step further, BERT combines the two sentences using special tokens at required places and then applies bi-directional cross-attention on the embedded input. Internally the model represents the inputs by combining the token embeddings along with position and segmentation embeddings.

It uses WordPiece embeddings along with three special tokens– [CLS], [MASK] and [SEP]. The [CLS] token stands for classification and is considered in classification tasks. The [MASK] token is used for masking the words randomly in the input, whereas the [SEP] token marks the end of first sentence and start of second in NSP task.

Fine tuning of BERT is relatively easy as the pre trained model remains as it is but only the required parameters as per the task are extracted and set accordingly. After pre-training, the specific output is fed to the output layer of the specific task. That is, the [CLS] representation is fed to output layer for classification task, similarly question answering task is provided with token representations.

The paper described various NLP tasks and their results generated by fine tuning the BERT model. Among these tasks are General Language Understanding Evaluation Benchmark (GLUE), consisting of diverse NLP tasks like language inference, question equivalence, linguistical acceptance of sentences, semantical equivalence of sentences, classification, sentence similarity, etc. BERT achieved a GLUE score up 80.5% with an absolute improvement of 7.7%. Similarly, it achieved 1.5% and 5.1% of improvement on SQuAD v1.1 and v2.0, for question answering NLP task.

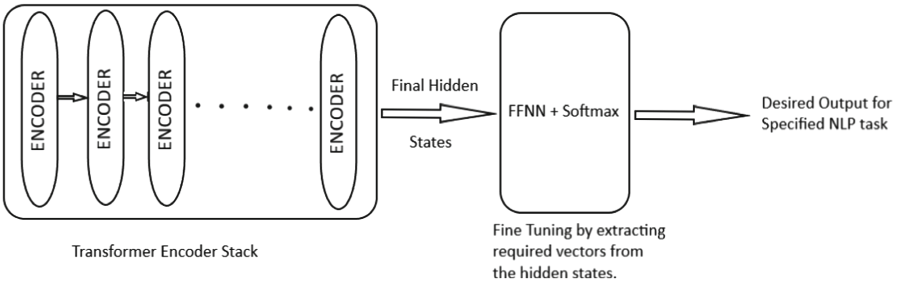

Fig. 1. Proposed BERT model.

2.1 Proposed General Architecture

After analysis of how BERT is fine tuned to carry out said tasks, we can represent the general architecture for BERT as shown in Fig. 1.

Consider, the model is applied to GLUE for classification task. Then only the hidden layer representing the classification token [CLS] vector is taken and a few additional parameters like weights are added, in order to feed it to the feed-forward neural network and softmax, which in turn normalizes the output required.

When SQuAD v1.1 is considered, the task is to determine whether the answer to the question lies in the passage or not. SQuAD v2.0 is experimented on same basis as SQuAD v1.1, except that an additional [CLS] token is added between the start and end tokens.

In case of Situations with Adversarial Generations (SWAG), where the task is to predict next sentence, again the [CLS] token is normalized, but the input sequence consists of 4 statements that can be possible continuation for the first sentence.

3 Recent Works in Implementation of BERT

BERT is the talk of the town ever since it was first put forward. It has attracted a lot of scientists to try and implement it on wide range of NLP tasks. In recent years, many experiments have been carried out on problems like – text summarization, automated grading system, text similarity score prediction, enhanced sentiment classification, reranking. Researchers are not restricting their experiments to English language either. Attempts are made to apply the above-mentioned tasks to different languages including various Indian languages.

3.1 Sentiment Classification

Yuxiang Zhou, Lejian Liao, Yang Gao, Rui Wang, and Heyan Huang in [2] suggested an enhanced model of BERT dealing with topic recognition at corpus level. The two models suggested here are- TopicBERT-ATP and TopicBERT-TA. The focus was on developing a model that was capable of capturing the essence of text on corpus level, rather than word and sentence level.

BERT is trained on Wikipedia and BookCorpus, thus when dealing with reviews or description of electronics equipment like washing machine or refrigerators, BERT often lacks because of homonymy or polysemy of words. Homonymy is where same words convey different meanings depending on the context whereas polysemy is when a word with same meaning conveys different sentiments. Moreover, neural networks using attention prioritize prominent signal words like "good" or "awful" more than words like "crowded" that grant more sentiments.

With all these in mind, the proposed solution was to enable BERT to capture the global sentiments and semantics of the text and accordingly classify the sentiments with more accuracy than existing models.

TopicBERT variants incorporate aspect term sentiment classification (ATSC), which is a classification task to predict nature of sentiments [positive, negative or neutral]. The TopicBERT-ATP (aspect topic prediction) fine-tunes BERT using ATSC in order to make it recognize the topic of the entire text corpus. The second variant TopicBERT-TA also uses ATSC but also adds a topic augmentation layer that is a low-level-representations that are unlabeled and change dynamically. These topic-representations take into consideration both the problems of homonymy and polysemy.

3.2 Text Summarization

The text summarization task can be said as the combination of text recognition and text generation tasks. Summarization also needs to deal with major information loss.

In order to generate a model that recognizes a topic relatedness as well as coherence, a model that is aware of the topics and functions in abstractive as well as extractive way, T-BERTSum was proposed by Tinghuai Ma, Qian Pan, Huan Rong, Yurong Qian, Yuan Tian, and Najla Al-Nabhan in [3].

The proposed model makes use of BERT along with neural topic model (NTM) and Long short term memory (LSTM). Extending the position, token and segment embeddings, T-BERTSum adds topic-aware sequences to enable topic-awareness hence generating topic embeddings. NTM is helpful for generating detailed topic embeddings. Once the output is generated, LSTM layers, that are stacked upon the output, are responsible for determining whether the generated summaries are from summaries of extractive models or not. Using such language model, a two-stage prototype is prepared that generates a summary with least information redundancy. The two stages take care of extractive as well as abstractive summaries.

3.3 Sentence Similarity Metrics for Indian Languages

Indian languages follow different grammar rules than English language. While evaluating similarity of sentences, several parameters are to be considered like synonyms, morphological variations as well as the context. Sentence-BERT (SBERT) [4], was proposed as a computationally better version of BERT and generated progressive results while analyzing similarity of text based on semantics.

[5] made use of sentence-BERT for evaluating similarity of text in two languages. The analysis was done on English-to-Hindi and English-to-Tamil machine translation

systems. Initially a reference sentence and a candidate sentence would be supplied to the model. The sentence-BERT based Similarity (SBSim) metric will then generate a similarity score on sentence level. Para-BERT, a model trained on large set of paraphrase sentences and multi-BERT are used for tokenizing the text in Hindi and Tamil languages, and their comparison is done. It was seen para-BERT generated less number of tokens but the generated tokens where larger in size as compared to multi-BERT.

The SBSim model described can successfully represent multilingual sentence in form of a vector of fixed size. It can opportunely classify between good and bad versions and can be applied to other Indian languages as well.

3.4 BERT-Based Automatic Short Answer Grading System

Intelligent tutoring systems have widely benefitted from Automatic Short Answer Grading (ASAG) systems, that analyzes the text and grade it accordingly with questions. As the name suggests, it analyses short answers and thus is trained on a small corpus of text, which affects its accuracy in grading. The questions itself consists of fewer prominent tokens representing the context.

BERT can be useful in this system as it can compute the exact context of the questions as well as the answers to increase the grading accuracy. As proposed in [6], in a BERT-based deep neural network model for ASAG task, the output of BERT is refined using a bidirectional LSTM. Thus, it useful in extracting global information along with the local context.

This new framework achieves good outcome because BERT captures the context and is again verified by bidirectional LSTMs. Different students can use various text and semantics for answering the same questions. Many times, the answer submitted may be completely irrelevant to the question asked. To identify such cases, the fine semantics of classic neural networks when combined with the ability, to capture deep semantics of even dynamic words of BERT, results in a cutting-edge model for ASAG.

3.5 Passage Ranking with BERT

When query-based tasks like question-answering are considered, the information retrieval mechanism focuses on ranking the documents/answers along with searching for pertinent output, from a large data corpus. Each of the documents generated as output for the user query, are scored in order to rank them according to their applicability.

In [7], BERT is implemented as a model to re-rank the passages that are retrieved as output. The paper executed the task by feeding the query/question as one sentence and the entire passage as another sentence in such a way that only the essence of the text is kept and question and passage size is reduced to 64 tokens and 512 tokens respectively, the special tokens inclusive. Finally, the score of the appositeness of the particular passage is determined by using the [CLS] token vector.

The paper discusses fine tuning BERT for re-ranking by calculating the cross-entropy loss between the relevant and irrelevant passages. The information retrieval (IR) mechanism applied is BM25. The datasets used for pretraining BERT as a re-ranker were MS MARCO, TREC-CAR. MS MARCO consist of user fetched top-10 answers from

Bing whereas TREC-CAR is made up of Wikipedia title concatenated with its text and subheadings. As BERT$_{\text{LARGE}}$ is used, that is originally trained on entire Wikipedia, the pre-training of BERT as a ranker was done on only 50% of the TREC-CAR training set. Nevertheless, BERT as a re-ranker achieved futuristic results on both of the training sets.

3.6 BERTweet Model for English Tweets

Tweets generally vary in its feature from the normal Wikipedia text or article available on internet. Tweets have a power to convey a complete thought process of user with less words. Moreover, the informal language/grammar, slangs and hashtags, that clearly indicates the context of the Tweet, may be discarded when traditional language models are applied, thus making the analytical tasks on Tweets data more complicated [8].

[9] proposed a model, with same architecture as BERT$_{\text{BASE}}$ and pre-trained using RoBERTa, named BRETweet for performing NLP tasks on Tweet data. The training set used consisted of 850M tweets resulting in 80 GB of data. The text of the tweets was tokenized first including the emoticons and converting them to text. The pre-training of BERTweet also took the usernames that were tagged and web links into consideration when normalizing them. Moreover, it also contemplated the retweeted, too short or too big tweets. This trained language model was then implemented on NLP tasks like Parts-Of-Speech (POS) tagging, text classification and name-entity recognition (NER) with the Tweets data at the focal point.

The datasets used for evaluating BERTweet for POS tagging were Ritter11T-POS, ARK-Twitter and TWEEBANK-v2 whereas for text classification were SemEval2017 Task 4A and SemEval2018 Task 3A, which are datasets for 3-class sentiment analysis and 2-class irony detection, respectively. The results were then compared with previous state-of-the-art models RoBERTa$_{\text{base}}$ and XML-R$_{\text{base}}$, and BERTweet out performed these models in every task and on every dataset used for experimentation. For the entire experiment, only English tweets were considered.

3.7 Applying BERT in E-commerce

BERT has shown the ability to learn the accurate context of the language and this ability can be useful coming to E-commerce tasks like product reviews or product segregation/association based on description and many more. However, the E-commerce data is not same as that of internet text articles and Wikipedia data. When dealing with E-commerce tasks the model must have a proper understanding of the product, its features, positive and negative characteristics of the product features and its overall domain. The basic BERT fails to capture this product domain knowledge in regard to E-commerce tasks.

[10] proposed a variant of BERT named E-BERT, which is a pre-training architecture that incorporates the domain knowledge in BERT. This domain knowledge mainly trains the model to identify the phrase level – the understanding about how the product features are phrased in the description and reviews – and product level knowledge which is of utmost significance in E-commerce tasks like product classification, review/aspect sentiment classification and so on. Two interesting approaches introduced here are Adaptive Hybrid Masking (AHM) and Neighbor Product Reconstructions (NPR). Adaptive Hybrid Masking (AHM) attempts to integrate the phrase level knowledge in the model.

AHM can be said as a modification to existing MLM, that works in two ways. First is it simply masks the words – word masking mode – that helps understand the word level semantics, while the second is phrase masking mode where AHM randomly masks domain phrases to extract the phrase semantics. AHM then calculates the losses incurred during learning and can switch between the two methods. The phrase pool focused on E-commerce was built as the resource for pre-training so as to apply AHM. To include the product level knowledge, NPR approach was used, based on the product association graph. It implemented cross attention, so that the model was able to learn about the product details, thus incorporating product level semantics.

Finally, the E-BERT was fine-tuned for following tasks: review-based question answering, aspect sentiment classification, product classification and aspect extraction. When compared with other variants of BERT, E-BERT-AHM was seen to perform better as well as the E-BERT language model. The Amazon QA dataset along with Amazon product metadata and SemEval Task 4 laptop dataset were used for pretraining purpose.

3.8 DistillBERT a Distilled Version of BERT

The BERT model proposed originally pretrained on Wikipedia text corpus and BookCorpus which are pretty huge and thus it was very expensive pre-training process. Furthermore, to use BERT, it was necessary to satisfy its memory requirements in order to implement it to any other NLP tasks. Originally, BERT training required 64 TPUs for over 4 days on Wikipedia (2.5B words approx.) and Google's BookCorpus (800M words approx.).

[11] applied the knowledge distillation technique on BERT LM. It is a method of transferring the learnings from a large model to a smaller one by compressing it. It can be considered similar to training a small model or a student model to imitate the actions of the teacher model or the larger model. This training of student model is mainly based on the training loss that is masked modelling loss and an added cosine embedding loss in case of DistillBERT- the student model. For developing this student model from the teacher- BERT model, the number of layers in the original architecture was reduced by the factor of 2, meaning it was trained by considering alternate layers in the BERT model, whereas the normalization and linear layers were highly optimized.

DistillBERT was then implemented to the similar tasks like sentiment classification on IMDb dataset and question answering tasks (SQuAD v1.1). It was at par with the results of BERT for same NLP tasks, if not better. It is significant to note that distillBERT has 40% less parameters as compared to the BERT model and is successful in maintaining its language understanding abilities up to 97%. As compared to $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$, distillBERT proved to be 60% faster.

4 Conclusion

In this paper, we reviewed the BERT model and its working, along with its implementations in various problem-statements. We reviewed how BERT can enhance the abilities of various model working on identifying and understanding natural languages. With a little fine-tuning on BERT model, it can be applied to various already existing frameworks to amplify their precision. When considering the learning ability of machines with respect to natural languages, there are various issues like grammatical semantics and meanings of the words along with variety of natural languages available. In this paper, we reviewed how BERT can be efficiently fine-tuned to tackle these issues. It also deals with homonymy, polysemy, hyponym and many other grammatical semantics of natural languages. However, BRET is a predefined model and its training is a lot more expensive. With smaller versions like DistillBERT- a faster equivalent, it is possible to apply BERT in a smaller environment. With its application in wide range of domain, BERT has improved the final results of many problem statements ranging from simple classification tasks to complex implementation in field of education and e-commerce.

References

1. Jacob Devlin, Ming-Wei Cheng, Kenton Lee, Kristina Tautonova: "BERT: Pre-training of deep bidirectional transformers for language understanding" in Proc. Annu. Conf. North Ameri. Chapter Assoc. Comp. Linguistics, Hum. Lang. Technol., 2019, p- 41714186.
2. Yuxiang Zhou, Lejian Liao, Yang Gao, Rui Wang, and Heyan Huang: "TopicBERT: Topic Enhanced Neural Language Model Fined-Tuned for Sentiment Classification", IEEE Transactions on Neural Networks and Learning Systems, 2021.
3. Tinghuai Ma, *Member, IEEE*, Qian Pan, Huan Rong, Yurong Qian, Yuan Tian, and Najla Al-Nabhan: "T-BERTSum: Topic Aware Text Summarization Based on BERT", IEEE Transactions on Computational Social Systems, Vol. 9, No. 3, 2022.
4. Nelis Reimers, Iryna Gurevych: "SentenceBERT: Sentence Embeddings using Siamese BERT-Networks."
5. K. Mrinalini, P. Vijayalakshmi, T. Nagaranjan: "SBSim: A Sentence-BERT Similarity-Based Evaluation Metric for Indian Language Neural Machine Translation Systems", IEEE/ACM Transactions on Audio, Speech and Language Processing, Vol. 30, 2022.
6. Xinhua Zhu, Han Wu, and Lanfang Zhang, "Automatic Short Answer Grading via BERT based Deep Neural Networks", IEEE Transactions on Learning Technologies, Vol. 15, No. 3, 2022.
7. Rodrigo Nogueira, Kyunghyun Cho: "Passage Re-Ranking with BERT"
8. V. Chakkarwar and S. Tamane: "Social Media Analytics during Pandemic for Covid19 using Topic Modeling," 2020 International Conference on Smart Innovations in Design, Environment, Management, Planning and Computing (ICSIDEMPC), 2020, pp. 279-282, doi: https://doi.org/10.1109/ICSIDEMPC4902s0.2020.9299617.
9. Dat Quoc Nguyen, Thanh Vu and Anh Tuan Nguyen: "BERTweet: A pre-trained language model for English Tweets"
10. Denghui Zhang, Zixuan Yuan, Yanchi Liu, Fuzhen Zhuang, Haifeng Chen, Hui Xiong: "E-BERT: Adapting BERT to E-commerce with Adaptive Hybrid Masking and Neighbor Product Reconstruction"
11. Victor Sanh, Lysandre Debut, Julien Chaumond, Thomas Wolf: "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter"