Reproduction and Application of ALBERT Model

Jianan Zhou
19210980081
International Business
School of Data Science
Fudan University
jnzhou19@fudan.edu.cn

Hanxiu Li
19210980085
International Business
School of Data Science
Fudan University
hxli19@fudan.edu.cn

Yuhao Chen
19210980073
Applied Statistics
School of Data Science
Fudan University
yhchen19@fudan.edu.cn

Abstract

With the application and development of pre-training model in natural language processing, machine reading comprehension no longer simply relies on the combination of network structure and word embedding. This paper briefly introduces the concepts of machine reading comprehension and pre-training language model, summarizes the research progress of machine reading comprehension based on ALBERT model, analyzes the performance of the current pre-training model on the relevant data set.

1 Introduction

Machine Reading Comprehension (MRC) is a task that tests a Machine's understanding of natural language by asking it to answer questions in a given context. Early MRC systems date back to the 1970s, the most famous of which is the QUALM system proposed by Lehnert. But the system that was being built was very small and limited to hand-coded scripts, making it difficult to generalize to a wider domain. Research into MRC was largely ignored in the 1980s and 1990s. In the late 1990s, Hirschman et al. proposed the Deep Read system and created a reading comprehension dataset consisting of 120 stories from third-to sixth-grade sources. Use these to build and evaluate a baseline system that includes a rule-based word bag approach and additional automated language processing (stem extraction, name recognition, semantic class recognition, pronoun resolution, and so on).These systems were able to retrieve correct sentences with 30 to 40 percent accuracy. Because these systems rely on a large number of manual rules or features, resulting in weak generalization, performance can degrade dramatically on large data sets containing more types of articles.

Due to the small size of the data set previously proposed and the limitations of rule-based and machine-based approaches, early MRC systems performed poorly and were difficult to use in practical applications. Since 2015, with the development of neural network, after preliminary training of word vector technology (such as Word2Vec, Glove, Fast Text, etc.) have also made certain progress, said of words and articles have made a lot of ascension, the other Seq2Seq structure of neural network and Attention mechanism is put forward that makes it possible to design more complex network model. DeepMind researcher Hermann et al. proposed a novel and inexpensive solution for creating large-scale supervised training data for learning to read comprehension models, constructing the CNN/Daily Mail data set. They also put forward the LSTM model based on attention the attentive reader, the impatient reader, they are in the very great degree is superior to the traditional Natural Language Processing (NLP) method.

In 2016, Chen et al. studied the above data set and proved that a simple and well-designed neural network model could improve the performance of CNN data set to 72.4% and Daily Mail data set to 75.8%.Compared with the traditional feature-based classifiers, the neural network model can better recognize word matching and definition. However, due to the data creation method and some other errors, the noise of the data set limits the further development.

To address these constraints, Rajpurkar et al., Stanford Question Answering Dataset (SQuAD) Dataset. The data set contains 100,000 questions from 536 Wikipedia articles, each answered with the text of the corresponding reading paragraph. This dataset is of high quality and can be reliably evaluated, making it a central benchmark for the field. Promoted a series of new reading comprehension models. By October 2018, the best-performing single BERT model system had achieved 91.8% of F1 value, surpassing the human performance of

91.2%.

With the appearance of GPT, BERT and other pre-training models, as well as the powerful improvement in reading comprehension tasks, a number of excellent pre-training language models have been born. After SQuAD data set, Stanford University released SQuAD2.0 data set, adding some questions with no answers compared with before and increasing the difficulty. Microsoft Research Asia released MSMARCO, a data set derived from real application scenarios, and Domestic Baidu released DuReader, a Chinese data set based on Baidu search and Baidu knowledge. In recent years, reading comprehension data sets in law, military and other aspects have emerged in China, which broadens the application of reading comprehension and also increases the difficulty. For machine reading comprehension with unanswerable questions (the type this article focuses on), the model needs to be capable of two aspects: 1) determining whether the question is answerable;2) Answer questions accurately. In order to make answerability judgment, it is necessary to have a deep understanding of the given text and a sophisticated discriminating design, and make the reading comprehension system closer to the practical application. Therefore, the research objective of this paper is to effectively solve the unanswerable questions while accurately answering the questions given in the paper. The ALBERT model, developed by Google Research, solves these two problems effectively.

2 MODELING

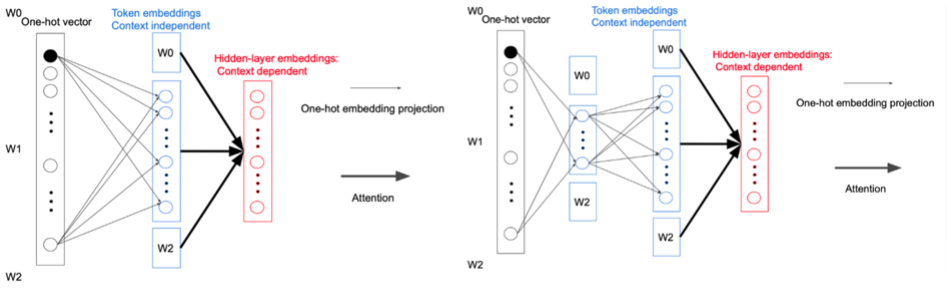
These advances in the field of language representation learning indicate that large models are extremely important for achieving SOTA performance. It has become a common practice to pretrain large models and refine them into smaller models in practical applications. Given the importance of model size, the researchers asked a question: Is it as easy to build a better NLP model as it is to build a larger one? The difficulty is that the memory available on the hardware is limited. Given that current SOTA models often contain hundreds of millions or even billions of parameters, scaling the model is subject to memory constraints. The researchers also observed that simply increasing the hidden layer size of models such as Bert-Large also led to performance degradation. The researchers doubled the hidden layer size of Bert-Large, and the accuracy of the model (Bert-XLarge) in the RACE benchmark was significantly reduced. To solve the above problems, Google researchers designed "A Lite BERT" (ALBERT) with far fewer elements than the traditional BERT architecture.

ALBERT overcame major obstacles to an extended pre-training model with two parameter reduction techniques. The first technique is to factor the embedded parameterization. The large lexicon embedding matrix is decomposed into two small ones so as to separate the size of the hidden layer from the size of lexicon embedding. This separation makes it easier to add hidden layers and does not significantly increase the number of arguments that the word is embedded with. The second technique is cross-layer parameter sharing. This technique prevents the number of arguments from increasing with the depth of the network. Both technologies significantly reduced the number of BERT's parameters without significantly affecting its performance, thus improving the parameter efficiency. The configuration of ALBERT is similar to BERT-Large, but the number of participants is only 1/18 of the latter, and the training speed is 1.7 times of the latter. These parametrization techniques can also act as a form of regularization, making training more stable and conducive to generalization.

To further improve ALBERT's performance, the researchers also introduced a self-monitored loss function for sentence level prediction (SOP).SOP mainly focuses on sentence coherence and is used to solve the problem of low efficiency of the next sentence prediction (NSP) in BERT. Based on these designs, ALBERT was able to scale to larger versions, with fewer arguments than Bert-Large, but with significant performance improvements. The researchers got new SOTA results on the well-known GLUE, SQuAD, and RACE natural language understanding benchmarks: RACE had an accuracy rate of 89.4%, GLUE had a score of 89.4, and SQuAD 2.0 had an F1 score of 92.2.

2.1 Factorized Embedding Parameterization

In the BERT model, word embedding and Encoder's output embedding dimensions are both 768. However, ALBERT believed that embedding at the word level is non-context-dependent, and the output value of the hidden layer not only includes the meaning of the word itself, but also includes some context information. So we should have $H >> E$, and the dimension of ALBERT's word vector is smaller than the dimension of the output value of



BERT Model ALBERT Model

Figure 1: Structural comparison between BERT and ALBERT models

encoder. In NLP tasks, the dictionary is usually very large, and $E \times V$ is the size of embedding matrix. If $H = E$ is embedding matrix as BERT, the number of participation will be very large, and in the process of backward communication, the updated content will be sparse.

Combining the above two points, ALBERT adopted a factorization method to reduce the number of parameters. One-hot vector is first mapped to a low-dimensional space with a size of E, and then mapped to a high-dimensional space. This means embedding matrix with a very low dimension is transformed into the space of the hidden layer through a high-dimensional matrix, and the number of parameters is reduced from $O(V \times H)$ to $O(V \times E + E \times H)$. The number of parameters decreased significantly when $E << H$.

The figure below is an experimental result of E selecting different values. Note that setting E to 768 is better when parameter sharing optimization scheme is not adopted, and setting E to 128 is better when parameter sharing optimization scheme is adopted.

Model	E	Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
ALBERT base not-shared	64	87M	89.9/82.9	80.1/77.8	82.9	91.5	66.7	81.3
	128	89M	89.9/82.8	80.3/77.3	83.7	91.5	67.9	81.7
	256	93M	90.2/83.2	80.3/77.4	84.1	91.9	67.3	81.8
	768	108M	90.4/83.2	80.4/77.6	84.5	92.8	68.2	82.3
ALBERT base all-shared	64	10M	88.7/81.4	77.5/74.8	80.8	89.4	63.5	79.0
	128	12M	89.3/82.3	80.0/77.1	81.6	90.3	64.0	80.1
	256	16M	88.8/81.5	79.1/76.3	81.5	90.3	63.4	79.6
	768	31M	88.6/81.5	79.2/76.6	82.0	90.6	63.3	79.8

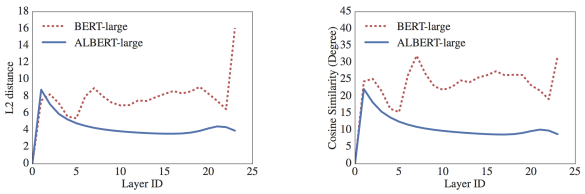
Figure 2: Result of E selecting different values

2.2 Cross-layer Parameter Sharing

ALBERT model also puts forward a method of parameter sharing, Shared parameters in Transformer has a variety of solutions, only full connection sharing layer or only share attention, and ALBERT is a combination of the above two kinds of schemes, all connection layer with attention to Shared parameters, that is to say, share the all the parameters inside the encoder, under the same order of magnitude of the Transformer by using this scheme after the effect is actually falling, but the decrease in the number of arguments a lot, training speed has improved a lot.

The following figure shows a comparison between BERT and ALBERT. Taking base as an example, BERT's parameter is 108M, while ALBERT is only 12M, but the effect does decrease by two points compared with BERT. Because of its high speed, BERT-XLarge was taken as the reference standard, whose parameter was 1280M. It was assumed that its training speed was 1. The training speed of ALBERT-XXLarge version was 1.2 times that of BERT-XLarge, and its parameter was only 223M.

In addition to the training speed mentioned above, ALBERT's output embedding of each layer has a smaller impact than BERT. The figure below shows the output distance between the different layers of L2 and cosine, so you can see that parameter sharing actually has the effect of stabilizing network parameters.



Model		Parameters	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg	Speedup
BERT	base	108M	90.5/83.3	80.3/77.3	84.1	91.7	68.3	82.1	17.7x
	large	334M	92.4/85.8	83.9/80.8	85.8	92.2	73.8	85.1	3.8x
	xlarge	1270M	86.3/77.9	73.8/70.5	80.5	87.8	39.7	76.7	1.0
ALBERT	base	12M	89.3/82.1	79.1/76.1	81.9	89.4	63.5	80.1	21.1x
	large	18M	90.9/84.1	82.1/79.0	83.8	90.6	68.4	82.4	6.5x
	xlarge	59M	93.0/86.5	85.9/83.1	85.4	91.9	73.9	85.5	2.4x
	xxlarge	233M	**94.1/88.3**	**88.1/85.1**	**88.0**	**95.2**	**82.3**	**88.7**	1.2x

Figure 3: Performance comparison between BERT and ALBERT

2.3 Cross-layer Parameter Sharing

BERT's NSP task is actually a dichotomous task. The positive sample of training data is by sampling two consecutive sentences in the same document, while the negative sample is by adopting two sentences in different documents. The main purpose of this task is to improve the effect of downstream tasks, such as NLI natural language reasoning task. However, subsequent studies found that the task was not effective, mainly because it was too simple. NSP actually contains two subtasks, topic prediction and relational consistency prediction, but topic prediction is much simpler than relational consistency prediction, and it actually has type effects in MLM tasks.

Predicting model contains subject is because is the sample is selected in the same document, the negative samples is selected in different document, if there are two documents, one is the entertainment related, one is related to politics, then the content of the negative sample selection is different themes, and positive samples in the entertainment in the document you choose prediction is the entertainment, the theme of the choice in political documents is the theme of the latter.

In the ALBERT model, in order to retain only the consistency task and remove the influence of topic recognition, a new task Sentence -Order prediction (SOP) was proposed. The positive sample of SOP could be obtained in the same way as that of NSP, and the negative sample could reverse the order of the positive sample. The SOP is chosen because it is based on the same document and focuses only on the order of sentences without subject-matter implications. And SOP can solve the task of NSP, but NSP can't solve the task of SOP, the task added to the final result increased by 1

	Intrinsic Tasks			Downstream Tasks					
SP tasks	MLM	NSP	SOP	SQuAD1.1	SQuAD2.0	MNLI	SST-2	RACE	Avg
None	54.9	52.4	53.3	88.6/81.5	78.1/75.3	81.5	89.9	61.7	79.0
NSP	54.5	90.5	52.0	88.4/81.5	77.2/74.6	81.6	**91.1**	62.3	79.2
SOP	54.0	78.9	86.5	**89.3/82.3**	**80.0/77.1**	**82.0**	90.3	**64.0**	**80.1**

Figure 4: Performance on different dataset

3 CONCLUSION

ALBERT actually reduces the memory through parameter sharing, and the prediction stage still needs the same time as BERT. If the ALBERT- XXLarge is adopted, the prediction speed will actually be slower. ALBERT solved the problem of increasing speed during training, and if you really want to reduce the total amount of computation, it is a complex and difficult task. You cannot have your cake and eat it. However, ALBERT is also more suitable to use feature base or model distillation to improve the final effect.

Sparse attention or Block Attention are some of ALBERT's likely future optimizations, which really do reduce computation. Futhermore, there are more dimensional features that need to be captured by other self-supervised learning tasks.

4 APPLICATIONS

4.1 Aid Decision Making

Machine reading comprehension technology, as an important research direction in the field of artificial intelligence, mainly aims at enabling machines to read texts, understand semantics, mine and deduce key information, and help humans to obtain required knowledge from massive fragmented information. In the era of data explosion, it is a challenging task to quickly obtain information that users care about and help them make decisions. Especially for professional fields, such as doctors' diagnosis records, historical data of the financial industry, court judgments, etc., users are required to have more relevant knowledge reserves, require a lot of preparatory work, and consume a lot of human and material resources.

In the knowledge-enabled information age, the machine reading and understanding can quickly collect information that users care about, help users analyze and solve problems, and give reasonable Suggestions. Therefore, introducing machine reading comprehension into a professional field can help users make better decisions and work more effectively.

4.2 Community-based Question Answering

With the rapid development of network technology, the scale of user-generated content in the Internet is constantly increasing. As a new network information resource, the research and application value of high-quality user-generated content is gradually emerging. QA pairs composed of questions and their answers are typical representatives of user-generated content and direct products of internet-mediated knowledge sharing among users.

Traditional community QA based on retrieval to match the user's main problem and find the answer, so the answer often contain a lot of noise, and by combining with machine reading comprehension

can contain a lot of noise from information online community content automatic recognition and extraction of QA information, quick to accurately obtain the concern of the answer.

4.3 Chatterbot

Intelligent chatbot is a program that simulates human conversations through natural language. It can provide external customer service and internal business assistance to achieve all-round efficiency improvement, cost reduction and efficiency increase.

At present, chatbots in specific scenes and fields have demonstrated high natural language understanding and processing capabilities, such as Xiao Du, Siri, Xiao Ai, etc. In addition to small talk, intelligent chatbots can also be used as QA robots to answer questions from the professional world. In the future, technology combined with machine reading comprehension will enable chatbots to more accurately identify users' questions and intentions.

5 REFERENCE

[1]Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[2]Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. Electra: Pre-training text encoders as discriminators rather than generators. In International Conference on Learning Representations.

[3]Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In ICLR.

[4]Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. ACL.

[5]Seohyun Back, Sai Chetan Chinthakindi, Akhil Kedia, Haejun Lee, and Jaegul Choo. 2020. NeurQuRI: Neural question requirement inspector for answerability prediction in machine reading comprehension. In ICLR.

[6]Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. Attention-overattention neural networks for reading comprehension. ACL.

[7]Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. 2017. Gated-attention readers for text comprehension. ACL.

[8]Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. ACL.

[9]Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. NeurIPS.