ReCAM@IITK at SemEval-2021 Task 4: BERT and ALBERT based Ensemble for Abstract Word Prediction

Abhishek Mittal **Ashutosh Modi**
Indian Institute of Technology Kanpur (IIT Kanpur)
amittal@iitk.ac.in
ashutoshm@cse.iitk.ac.in

Abstract

This paper describes our system for Task 4 of SemEval-2021: Reading Comprehension of Abstract Meaning (ReCAM). We participated in all subtasks where the main goal was to predict an abstract word missing from a statement. We fine-tuned the pre-trained masked language models namely BERT and ALBERT and used an ensemble of these as our submitted system on Subtask 1 (ReCAM-Imperceptibility) and Subtask 2 (ReCAM-Nonspecificity). For Subtask 3 (ReCAM-Intersection), we submitted the ALBERT model as it gives the best results. We tried multiple approaches and found that Masked Language Modeling(MLM) based approach works the best.

1 Introduction

Computers' ability to understand, represent, and express text with abstract meaning is a fundamental problem towards achieving true natural language understanding. In past decades, significant advancement has been achieved in representation learning. SemEval-2021 Task 4 : Reading Comprehension of Abstract Meaning (ReCAM) (Zheng et al., 2021) explores the ability of machines to understand abstract concepts and proposes to predict abstract words just as humans do while writing article summaries. In the shared task, text passages are provided to read and understand abstract meaning. It consists of three subtasks where the first two subtasks are based on two different definitions of abstractness 1) Imperceptibility (Spreen and Schulz, 1966) and 2) Non-specificity (Changizi et al., 2008) and the third subtask discusses their intersection.

Many cloze-style reading comprehension datasets like CNN/Daily Mail (Hermann et al., 2015) and Children's Book Test (CBTest) dataset (Hill et al., 2016) and models (Dhingra et al., 2016; Munkhdalai and Yu, 2017) similar to this task exist, where a missing word has to be inferred. However, these previous datasets and models have mostly focused on inferring concrete words or concepts like named entities, but this task moves the focus from concreteness to abstractness of words in reading comprehension. This can prove to be quite useful for current ongoing research in the field of abstractive summarization.

We participated in all the three subtasks. We mainly used an ensemble of BERT and ALBERT as our final model for submission on subtasks 1 and 2. We were ranked 13th on Subtask 1 and 11th on Subtask 2. We submitted the ALBERT model on Subtask 3. All of our code is made publicly available on Github[1]. We approached this task in two ways. One is a Multiple Choice Question answering (MCQ) based approach and other a Masked Language Modeling (MLM) approach. Through experiments, we concluded that such tasks are best addressed using a masked language model.

The rest of the paper is organised as follows. Section 2 describes the problem statement formally and also gives a brief description of the dataset provided by the task organizers. Section 3 introduces the related work. Section 4 describes our proposed approach and Section 5 gives the experimental details. We enlist our results in Section 6 with a brief error analysis. Finally, we give concluding remarks in Section 7.

2 Background

2.1 Problem Description

A passage P, a followup question Q with a **@placeholder** and a list of candidate answer words $W = \{W_1, W_2, W_3, W_4, W_5\}$ are given as an input to the model. The task is to output the correct answer

[1] https://github.com/amittal151/SemEval-2021-Task4_models

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 175–182
Bangkok, Thailand (online), August 5–6, 2021.

Sr No.	Ambiguous Examples					
1	**Article** : [7 January 2016 Last updated at 10:26 GMT Robin Sollis packs it all away in his garage - leaving no room for his car. His is one of 14 homes on a Cirencester street which have raised ˆA £ 1,200 for charity with their displays this year. Tracey Miller reports.]					
	Question : Taking down this @placeholder Christmas lights display in Gloucestershire will take its owner several days.					
	Options	Impressive	Special	Annual	Fake	New
	There is nothing about Gloucestershire in the article !					
2	**Article** : [20 April 2017 Last updated at 00:28 BST The reason is that cafe and shop owners who provide wi-fi could be held liable for illegal activity that happens on their network,such as piracy or illegal downloads. But campaigners are trying to change this law. Videojournalist: Joe Miller.]					
	Question : For a country with a reputation as a tech powerhouse , public wi - fi is @placeholder difficult to find in Germany.					
	Options :	often	also	surprisingly	particularly	rather
	There are many plausible answers(rather is also feasible) to the question. Use of words may depend on personal choices.					

Figure 1: Few ambiguous examples in Subtask 1 Dev set (Option marked in green is the correct answer)

W_i from W ($W_i \in W$) by learning the function F such that $W_i = F(P, Q, W)$

The first two subtasks focus on the two different definitions of abstractness and the third subtask captures the relationship between the two views of abstractness. The evaluation metric for all three subtasks is the accuracy of the predictions made by the model. The subtasks are enlisted below :

1. **ReCAM-Imperceptibility** - Abstract words refer to ideas and concepts that are not immediately perceivable by our senses like culture, objective, etc.

2. **ReCAM-Nonspecificity** - According to this definition, abstract words refer to holistic terms, e.g., animal, body, etc.

3. **ReCAM-Intersection** - In the third subtask, the system needs to be trained on one definition of abstractness (Imperceptibility) and evaluated on the other (Nonspecificity) and vice-versa.

Task	Training	Dev	Test
1	3227	837	2025
2	3318	851	2017

Table 1: Number of examples in dataset

2.2 Data Description

The task organizers have provided training and validation dataset for Subtask 1 and Subtask 2. Each training and validation set example is in the form of a dictionary containing an article, a question and 5 options. One word in the question is missing and is represented by "@placeholder", and we have to predict the word out of the given 5 options.

The data set has English news articles and questions are constructed from the summaries of these articles. The data statistics are provided in table 1. The dataset poses two major challenges. Firstly, the passages are quite long. Their distribution is shown in figure 2 and 3. The long article length leads to a loss of context when we truncate the article in a transformer based model due to its max token length limits. Secondly, the dataset contains some ambiguous examples where more than one correct answer could be feasible or the question's context is missing from the article. Some examples are shown in the Figure 1.

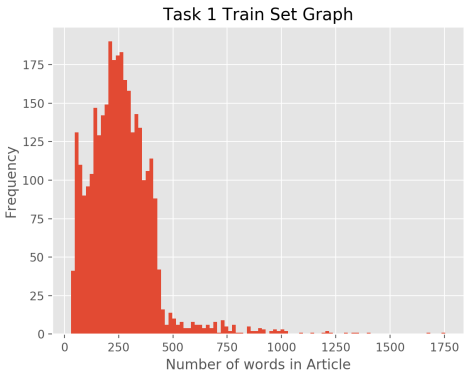

Figure 2: Task 1 Article statistics

3 Related Work

Much work has been done for the prediction of concrete words unlike ours where we need to predict abstract words in reading comprehensions. Gated Attention Reader (Dhingra et al., 2016) predicts missing concrete words in CNN/Dailymail datasets with a high accuracy. The attention mechanism plays a crucial role in recognizing which sections of the article are more important to answer the questions. Extracting context from the article is a vital

Figure 3: Task 2 Article statistics

part of the task. This task requires comprehensive natural language understanding, going beyond the meaning of individual words and sentences. We explored some of the pre-trained transformer models (Vaswani et al., 2017) as these capture the context better due to the self-attention mechanism. Moreover, pre-trained models are readily available.

The task is somewhat similar to a multiple choice question answering task. We experimented with the MCQ based approach as mentioned by Radford (2018) where a linear layer is built over the transformer and the correct answer is predicted by applying softmax over the probabilities of each option.

One approach to get context from the article is to extract the most relevant sentences to the question with sentence similarity techniques (Reimers and Gurevych, 2019). We experimented with this approach and extracted "Top-k" sentences that were most semantically similar to the given question from the article.

One of the major challenge in this task is to handle the long length of the article. Pappagari et al. (2019) discuss the approach of using hierarchical transformers for text classification problem to tackle long passages. BERT is applied to text segments and an LSTM layer or transformer is applied to get document embedding.

Another approach is to model the shared task as a masked language modeling task. The transformer based models like BERT (Devlin et al., 2018) and ALBERT (Lan et al., 2020) have been trained via the masked language modeling objective. BERT has also been trained on the Next Sentence Prediction task, and ALBERT has been trained on the Sentence Ordering task. In Lan et al. (2020), it is mentioned that Sentence Ordering task is a better way to understand the similarity and extracting context from two sentences and thus, ALBERT works better than the BERT model.

4 System Overview

We explored multiple models and methodologies. We first experimented using an encoder with an attention based approach. From their results, we observed that an MLM based approach would work better than an MCQ based approach.

Consequently, we tried BERT and ALBERT models and their ensemble with the MLM based approach. However, for comparison purposes, we also worked with the MCQ method and its system is described below along with our other approaches.

4.1 Encoders with Attention

4.1.1 Binary Classification with Attention

We tried a binary classification based approach where we give each option a score of being a correct answer. Our model consisted of two encoders, followed by a binary classifier. One encoder is for encoding the question and one for encoding the article. First, we feed the question into the question encoder which gives us the context vector of the question. Then, we feed the article along with the hidden weights from the question encoder into the article encoder and apply attention weights over them to find the context of question within the article. Finally, we input an option word, the hidden weights obtained from the article encoder and the context vector from question encoder into the binary classifier layer which gives us the score for the given option. The option word with highest score is predicted as our answer.

4.1.2 Cosine Similarity of predicted word with options

In this approach, we use the article and question encoders as described in the first approach to encode the article and question. However, instead of using a binary classifier, we used a decoder layer to predict the missing word. We used the "@placeholder" token's hidden embedding as an input to the decoder layer along with the context vector from question encoder and hidden weights from article encoder. This layer predicts a word from vocabulary which would fit in the place of "placeholder". We then compute this word's cosine-similarity with the given 5 options. The most similar option word is predicted as the answer. This method is similar

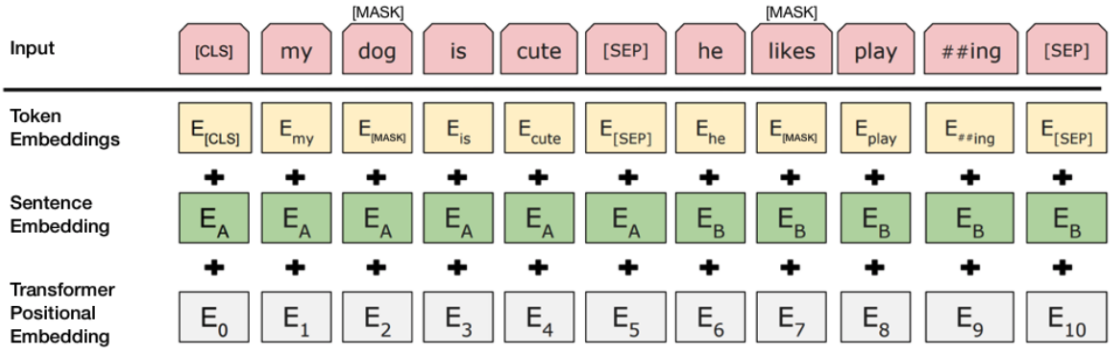

Figure 4: Masked Language Model (Image Src : (Devlin et al., 2018))

to an MLM based approach, and the first approach used above is somewhat similar to an MCQ based approach. This method gave slightly better results than the first approach, and thus, it gave us an idea that an MLM model should work better for our subtasks as compared to an MCQ based model.

4.2 Transformer based Models

4.2.1 Multiple choice Question Answering

The architecture of this approach is similar to that proposed by Radford (2018). We have a linear layer over a transformer model like BERT which takes the embedding of [CLS] token and calculates the cosine similarity of this token with given 5 options. Since, the [CLS] token represents the aggregate sequence representation (Devlin et al., 2018), it encodes the context of article and question together. The input sequence to the model is the concatenation of article and question (where the "@placeholder" is replaced by an option) delimited with the [SEP] token. On top of this, we have a softmax layer which calculates the score for each given option.

4.2.2 Masked Language Modeling

We used the transformer models like BERT and ALBERT for masked language modeling since they have been trained via the MLM objective. In this approach, the input sequence to our model is the concatenation of question and article tokens delimited with the [SEP] token where the "@placeholder" word in the question has been masked. We truncate the article from the end to fit into the maximum token sequence length. Since our task requires context reading from the article, we used different sentence embedding of the transformer models for question and article. An example of input sequence is given in Figure 4. The model's output is a probability vector with probability scores of replacing the masked token with any word in the vocabulary. We used the scores computed for the given 5 options and predicted option with the highest score as the correct answer.

We did an ensemble of BERT and ALBERT model predictions by taking the average score for each option predicted by these models. If the scores of BERT model predictions of the 5 options in a given example are B = $\{B_1, B_2, B_3, B_4, B_5\}$ and the scores of ALBERT model predictions are A = $\{A_1, A_2, A_3, A_4, A_5\}$, then our Ensemble model gives the scores

$$S = \{S_i : S_i = \frac{A_i + B_i}{2} \; \forall i \in \{1, 2, 3, 4, 5\}\}$$

We later also did an ensemble of two ALBERT models where one is fine-tuned on the given subtask, and the other one is not. This gave us the best results on Subtask 1 and Subtask 2. However, we tried this approach in the post-evaluation phase and thus, we did not submit this system on the leaderboard.

Model	Task 1	Task 2
GAReader (baseline)	0.251	0.243
Binary Classification with Attention	0.201	0.212
Cosine Similarity approach	0.256	0.249

Table 2: Results of Encoder Based Approaches on Dev sets (Metric : Accuracy)

5 Experimental Setup

Our implementation uses the PyTorch library (Paszke et al., 2019) for deep learning models and the Transformers library by HuggingFace (Wolf et al., 2020) for the pre-trained transformer models and corresponding tokenizers.

	Model	Task 1	Task 2	Task 3 (Train 1, Val 2)	Task 3 (Train 2, Val 1)
1	BERT (MCQ approach)	0.195	0.202	0.201	0.198
2	BERT	0.6654	0.6523	0.475	0.449
3	BERT_Large - without Article	0.681	0.690	0.5347	0.5364
4	BERT_Large	0.7455	0.7403	0.6451	0.5913
5	ALBERT xxlarge - without Article - Not Finetuned	0.7011	0.7262	0.7262	0.7011
6	ALBERT xxlarge - Not Finetuned	0.8172	0.8190	**0.8190**	**0.8172**
7	ALBERT xxlarge - Finetuned	0.8291	0.8345	0.7426	0.6977
8	Ensemble (4 + 6)	0.8375	0.8401	0.7826	0.7729
9	Ensemble (6 + 7)*	**0.8685**	**0.8554**	0.8143	0.8064

Table 3: Transformer models' results on Dev sets of each task (Metric : Accuracy)

* Modified System after submission and not submitted in the task

We first experimented with the baseline model, Gated-Attention reader (Dhingra et al., 2016) provided by the task organizers. This model did not give good results, as shown in Table 2.

Then, we experimented with our Encoder based approaches as described in Section 4. We experimented with various loss functions like NLL loss, MSE and CrossEntropy loss. But, the results were poor for these methods too (Table 2). However, the Cosine similarity based approach (Section 4.1.2) performed slightly better than the Binary classification with Attention approach (Section 4.1.1) indicating that an MLM approach should work better than an MCQ approach.

To verify our claim, we experimented with the BERT Base model with the MCQ approach, which gave quite less accuracy, no better than a random prediction. However, the BERT model with the MLM approach performed way better on all the subtasks. We experimented with both large and small variants of BERT and ALBERT models where the large variants performed better as expected. We fine-tuned both the models without freezing any layers with Adam optimizer (Kingma and Ba, 2017). We fine-tuned the BERT model for 3 epochs and ALBERT model for 1 epoch. We used the learning rate of 5e-5 and a max-sequence length of 256 for both BERT and ALBERT. We also used pre-trained ALBERT model without any fine-tuning in some of our experiments.

We also experimented with the input sequence to understand and compare the degree of context-reading done in ALBERT and BERT models. We changed the input sequence to contain only question tokens and then passed this sequence to our models. We then compared these results with the results obtained after passing the complete input sequence containing both article and question tokens. BERT gave an improvement of around 5-6% after passing the complete input sequence. However, ALBERT shows much more improvement of around 11-12% with the complete sequence. It shows that ALBERT 's training on a Sentence Ordering task is more effective for MLM tasks like ours than the BERT's training on Next Sentence Prediction task.

We then experimented with the ensemble of BERT Large and ALBERT xxlarge-v2 model predictions. We experimented by assigning different weights to BERT and ALBERT models and found out that equal weights to both works better. We later did an ensemble of the fine-tuned ALBERT model with a non fine-tuned ALBERT model. It gave much improved results on Subtask 1 and Subtask 2.

6 Results and Analysis

The results of all the transformer based approaches are given in Table 3. The recurrence based models did not work and predicted answers with a random probability. We used GloVe (Pennington et al., 2014) vector embeddings that are not contextualized, unlike BERT embeddings. Moreover, the task requires some world knowledge since we need to predict an abstract word whose meaning can possibly be encoded if it is trained on large English corpus. Transformer based models are trained on large corpora and implicitly learn concepts grounded in

Type	Example	
WC	**Article**	... ”Tennis chose me. It’s something I never fell in love with,” Tomic told Australia’s Channel Seven. ”Throughout my career I’ve given 100%. I’ve given also 30%. But if you balance it out, I think all my career’s been around 50%.” ....
	Question	Bernard Tomic says he has never ” really tried ” throughout his tennis career, adding that he has probably been @placeholder at “around 50% ”.
	Options	(A) held (B) aiming (C) honoured (D) operating (E) shown
	Scores	(A) 16.994 (B) 29.573 (C) 8.331 (D) 18.471 (E) 11.549
WN	**Article**	.... ”These are home games that we have to win,” McClaren said. ”We are not performing individually and collectively the way that we did up until the Leicester replay. Has that taken too much out of us? I don’t know. ”We are not getting the rub of the green and we were doing that before. We are not scoring the first goal and we are not scoring goals....
	Question	Manager Steve McClaren says Derby County ’s @placeholder have dropped and he has demanded an immediate response.
	Options	(A) chances (B) body (C) standards (D) side (E) artefacts
	Scores	(A) 28.372 (B) 7.169 (C) 27.527 (D) 10.246 (E) 8.395
CC	**Article**	Chiriac Inout was found in John Bright Street at about 23:30 GMT on 29 November, one of the coldest nights of the year. Police are investigating after CCTV appeared to show someone searching his pockets while he laid in a loading area behind The Victoria pub. An inquest date is yet to be fixed, the coroner’s office confirmed.
	Question	A coroner has named a rough sleeper who may have had property @placeholder before he died in Birmingham city centre .
	Options	(A) lost (B) collapsed (C) stolen (D) delays (E) flowers
	Scores	(A) 13.214 (B) 12.342 (C) 27.909 (D) 2.336 (E) 4.510
CN	**Article**	.... The Blue Peter team say that Lindsey is safe and on her way back to dry land. Sport Relief said: ”Lindsey’s Sport Relief challenge was always going to be incredibly hard and zorbing many miles across the Irish Channel is a huge achievement. .....
	Question	Blue Peter ’s Lindsey Russell has ended her attempt to cross the @placeholder between Northern Ireland and Scotland in a giant inflatable barrel for Sport Relief .
	Options	(A) gap (B) boundary (C) sea (D) title (E) difference
	Scores	(A) 24.295 (B) 26.728 (C) 26.874 (D) 4.482 (E) 18.486

Table 4: Some examples where our model makes mistakes or give correct results on Subtask 2 dev dataset. The options highlighted in blue are the correct answers and options highlighted in red are predicted by our model.

* WC - Wrong Confident - 27 such examples
* WN - Wrong Confused - 109 such examples
* CC - Correct Confident - 446 such examples
* CN - Correct Confused - 269 such examples

the world. Hence, transformer based approaches work better in our case.

The BERT model with the MCQ approach uses the embeddings of the ‘[CLS]’ token to predict the correct option. But, it doesn’t exploit the position of “@placeholder” token and hence it becomes difficult for the model to predict the correct result.

We observe that our Ensemble models work better on Subtask 1 and Subtask 2 as compared to the BERT and ALBERT models. However, on Subtask 3, the ALBERT model, which is not fine-tuned gives the best results. It owes to the fact that subtasks 1 and 2 differ a lot. If we fine-tune our model on one of the subtask, then it performs worse on the other.

For our final submission, we submitted our Ensemble model (8) (Table 3) for Subtask 1 and Subtask 2. We also submitted the fine-tuned ALBERT model on these subtasks. In Subtask 1, we are ranked 13th with our Ensemble model (8) with an accuracy of 0.8212 on test set. In Subtask 2, we are ranked 11th with the fine-tuned ALBERT model with an accuracy of 0.8761 on test set. Surprisingly, in Subtask 2, fine-tuned ALBERT model performed better than our Ensemble model on the test set. This is possibly because our ensemble system performed only marginally better than fine-tuned ALBERT model on dev set. In Subtask 3, we submitted the non fine-tuned ALBERT model.

For understanding the mistakes made by our submitted Ensemble system, we analysed the confidence scores of our model’s predictions. We performed the analysis of the system on the dev set of Subtask 2. We set a threshold factor (TF) of “1.4” for deciding between confident and confused predictions. If the confidence score of the model’s predicted option is P and the score for the correct option word is T, then the model is confident in its

predicted answer if the following condition holds :

$$P \geq TF * T$$

It turns out that the model makes 50% confident predictions out of all the predictions in the dev set. Also, the model makes 20% wrong predictions confidently, while in 80% of the wrong predictions, model is confused between two options. In many cases, it is unable to understand the context properly and in a few cases, the model lacks the necessary world knowledge (Table 4). In first example in Table 4, the model predicts 'aiming' as the answer quite confidently. However, 'operating' is a more appropriate option due to the context given in the article. This example shows that our system fails to read the context properly in some cases. Also, consider the second example in Table 4. Here, the model is confused between two options : 'chances' and 'standards'. Although, it is mentioned in the article clearly that Derby is performing poor in a past few matches, the model is not confident in predicting 'standards' which is the most suitable option here. It implies that our model is generally confused between all the option words that are semantically applicable in the question statement. Consider example 4 from Table 4. Here, the model is confused between option words 'sea' and 'boundary' because both of them fit well into the question. In order to make model more confident on such examples, we need to incorporate world knowledge into our system.

We also performed a similar post-competition analysis on our Ensemble System (9) (Table 3) and found out that it continues to make similar mistakes. But, in cases like example 1 in Table 4, it gives correct results. This is because, we used an ALBERT fine-tuned model instead of BERT fine-tuned model in this system which is better in context-reading as compared to BERT. Thus, this system gave slightly improved results.

7 Conclusion

The task of predicting abstract words with context from a question and article is quite novel in itself. We showed that this task can be modelled better as a masked language modeling task rather than multiple choice question answering task. The transformer based approaches worked best, where we used BERT and ALBERT models and their ensembles. These models are pretrained models and hence they perform better on our small dataset after fine-tuning. We were able to improve the results of ALBERT model with our Ensemble model on subtasks 1 and 2, but on Subtask 3, the ALBERT model performs better. In future, we shall try to improve our results on Subtask 3. In our current approaches, we haven't used options while training the model. We can try using a pairwise ranking loss function to rank the options according to their scores with a linear layer built on top of transformer models. This will help the model to predict answers more confidently and hence might also improve results. Moreover, we have used the same approach for Subtask 1 and 2. In future, we aim to incorporate common sense knowledge, for example, prototypical knowledge about activities in the form of scripts (Modi and Titov, 2014; Modi, 2016, 2017; Ostermann et al., 2018), or in the form of semantic networks like ConceptNet (Speer et al., 2018) for tackling two different definitions of abstractness and incorporating some knowledge in the two subtasks.

References

Mark Changizi, Andrew Hsieh, Romi Nijhawan, Ryota Kanai, and Shinsuke Shimojo. 2008. Perceiving the Present and a Systematization of Illusions. *Cognitive science*, 32:459–503.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805.

Bhuwan Dhingra, Hanxiao Liu, William W. Cohen, and Ruslan Salakhutdinov. 2016. Gated-Attention Readers for Text Comprehension. *CoRR*, abs/1606.01549.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching Machines to Read and Comprehend.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2016. The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization.

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.

Ashutosh Modi. 2016. Event embeddings for semantic script modeling. In *Proceedings of The 20th*

SIGNLL Conference on Computational Natural Language Learning, pages 75–83.

Ashutosh Modi. 2017. Modeling common sense knowledge via scripts.

Ashutosh Modi and Ivan Titov. 2014. Inducing neural models of script knowledge. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 49–57.

Tsendsuren Munkhdalai and Hong Yu. 2017. Reasoning with Memory Augmented Neural Networks for Language Comprehension.

Simon Ostermann, Ashutosh Modi, Michael Roth, Stefan Thater, and Manfred Pinkal. 2018. Mcscript: A novel dataset for assessing machine comprehension using script knowledge. *arXiv preprint arXiv:1803.05223*.

Raghavendra Pappagari, Piotr Żelasko, Jesús Villalba, Yishay Carmiel, and Najim Dehak. 2019. Hierarchical Transformers for Long Document Classification.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

A. Radford. 2018. Improving language understanding by generative pre-training.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2018. Conceptnet 5.5: An Open Multilingual Graph of General Knowledge.

Otfried Spreen and Rudolph W. Schulz. 1966. Parameters of abstraction, meaningfulness, and pronunciability for 329 nouns.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. HuggingFace's Transformers: State-of-the-art Natural Language Processing.

Boyuan Zheng, Xiaoyu Yang, Yuping Ruan, Quan Liu, Zhen-Hua Ling, Si Wei, and Xiaodan Zhu. 2021. SemEval-2021 task 4: Reading comprehension of abstract meaning. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*.