Masked Reasoner at SemEval-2020 Task 4: Fine-Tuning RoBERTa for Commonsense Reasoning

Daming Lu
Baidu Research
Sunnyvale, CA
USA
ludaming@baidu.com

Abstract

This paper describes the masked reasoner system that participated in SemEval-2020 Task 4: Commonsense Validation and Explanation. The system participated in the subtask B. We propose a novel method to fine-tune RoBERTa by masking the most important word in the statement. We believe that the confidence of the system in recovering that word is positively correlated to the score the masked language model assigns to the current statement-explanation pair. We evaluate the importance of each word using InferSent and do the masked fine-tuning on RoBERTa. Then we use the fine-tuned model to predict the most plausible explanation. Our system is fast in training and achieved 73.5% accuracy.

1 Introduction

Commonsense validation and explanation is a critical area in natural language understanding. Recent research advances (Yang et al., 2019; Devlin et al., 2018; Liu et al., 2019) pushed the bar in this area to a new height. SemEval 2020 Task 4 (Wang et al., 2020) is focused on this area. It has 3 subtasks. Subtask A is focused on commonsense validation. It gives two statements. One of them is against commonsense. For example,

1) He poured orange juice on his cereal.
2) He poured milk on his cereal.

The system needs to know which one is against commonsense. Subtask B is what we participated in. It first gives a statement, which is wrong or not proper. Then it lists 3 explanations. One of them can explain why the statement is not proper. For example,

statement: He poured orange juice on his cereal.
Explanations:
1) Orange juice is usually bright orange.
2) Orange juice doesn't taste good on cereal.
3) Orange juice is sticky if you spill it on the table.

The system needs to select it out of the three. Subtask C, in our opinion, is more challenging. The system needs to generate an explanation that explains why a statement is against commonsense. For example:

Statement: He put an elephant into the fridge.
Possible Valid Reasons:
1) An elephant is much bigger than a fridge.
2) A fridge is much smaller than an elephant.
3) Most of the fridges are not large enough to contain an elephant.

BLEU(Papineni et al., 2002) is used to judge whether the generated reason is a valid one. Subtask B is very similar to other famous datasets, such as CommonsenseQA(Talmor et al., 2018) and COPA(Roemmele et al., 2011). Many previous research(Talmor et al., 2018; Lin et al., 2019) showed that pre-trained language models can implicitly hold knowledge thus are able to answer these questions well. By fine-tuning based

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 411–414
Barcelona, Spain (Online), December 12, 2020.

on these pre-trained models, many datasets archived new heights. Our system fine-tuned in a novel way. We argue that whether a explanation fits a statement is strongly correlated to the 'core' of the statement, which should be the most important word. We first use InferSent to locate the most important word of the statement. Then we mask that word and start fine-tuning RoBERTa. By concatenate the proper explanation to the statement by adding a few concatenate words such as ' is wrong because ', we can get a full-text format. After the fine-tuning, we can apply the model to the test set. We provide our code on GitHub for reproduce purpose: `https://github.com/daming-lu/Code-for-SemEval2020-Task4`.

2 System Description

2.1 Problem Abstraction

Suppose we have an input statement $s = \left(w^{(1)}, w^{(2)}, \ldots, w^{(L_s)}\right)$, where each $w^{(i)}$ is a word. We also have a set of candidate explanations:

$$E = \left\{e_i = \left(e_i^{(1)}, e_i^{(2)}, \ldots, e_i^{(L_i)}\right)\right\}_{i=1\ldots n}$$

we aim to identify the most proper explanation $e^* \in E$ which can explain why statement s is improper. L_s is the length of the statement. The candidate explanations can have various lengths.

2.2 Core Word

In InferSent (Conneau et al., 2017), the authors focus on getting sentence embeddings that hold semantic information that are useful in natural language inference. We find that we can find the most important word, a.k.a the 'core' word in the statement, by representing the statement. We use both the statement and the candidate explanations as the corpus to build a relatively small vocabulary. The semantic information that implicitly stored in InferSent can help pinpoint the core word in the statement. See Figure 1 for more details.

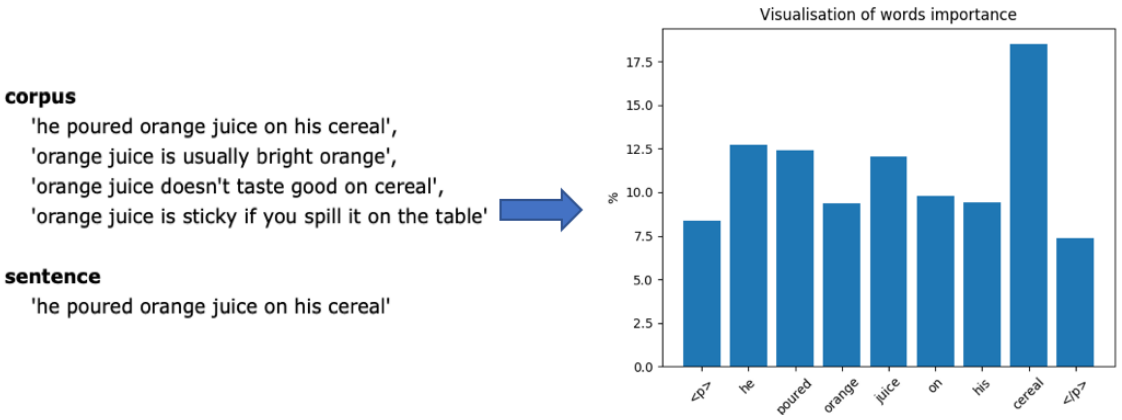

Figure 1: Example of a core word *'cereal'*

2.3 Sentence Evaluation

For each pair of $\langle s, e_i \rangle$, we first build its full-text format. We simply add *' is wrong because '* to concatenate s and e_i. For example, the 'cereal' example mentioned in Introduction will look like below:

He poured orange juice on his cereal is wrong because orange juice doesnt́ taste good on cereal.

We then mask the core word, i.e. *'cereal'*, in the statement and train the model to try to recover the masked word. Intuitively, the higher confidence the model has in recovering the masked word, the more

plausible the explanation e_i can best explain the statement s. We denote Sen as the concatenated full-text sentence. Following the notation in (Song et al., 2019), we note $Sen_i^{\setminus w}$ as the sentence Sen_i with the word w replaced by the [MASK] token. The sentence evaluation formula is as follows:

$$Score(Sen_i) = \log\left[P\left(p^{(k)}|Sen_i^{\setminus p^{(k)}}\right)\right]$$

where k is the index of core word. Masked word probability is estimated from a direct calculation on the pre-trained masked language model, in our case, RoBERTa. See Figure 2 for more information.

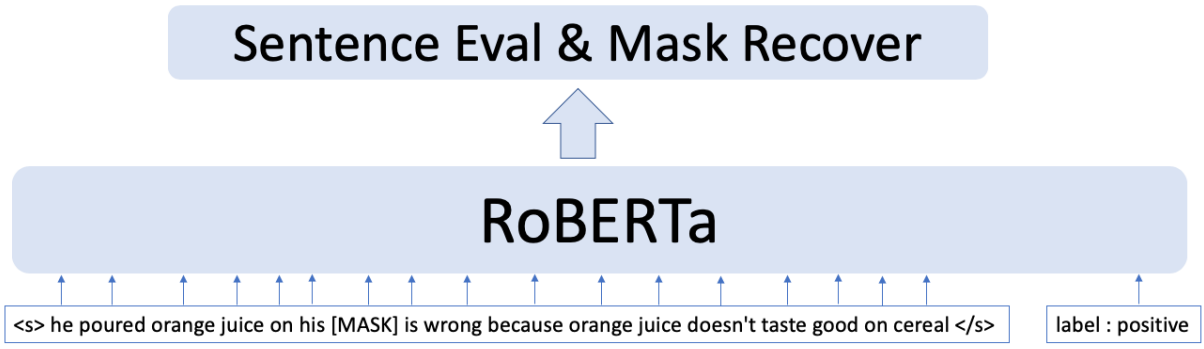

Figure 2: System architecture

3 Data

Trial data were given out in practice period. For Subtask B, trial data has 2,021 rows. Then 10,000 training data were released. Each has 1 improper statement and 3 candidate explanations. Follow concatenating method mentioned above, we have 30,000 training data for our system, 10,000 are positive and 20,000 are negative. We have 997 dev data that can be used as testing data during the practice period. Both trial, training and dev data have gold answers. Then in the evaluation period, the real test data were released whose gold answers are never revealed. See Table 1 for more information.

Table 1: Dataset Information

Dataset	Number of data records	have gold answer
Trial	2,021	Yes
Train	10,000	Yes
Dev	997	Yes
Test	1,000	No

4 Experimental Results

During the training period, our system reached 83.21% accuracy after 3 epochs, 85.22% after 6 epochs and 87.43% after 10 epochs. We trained on 2 Nvidia GEFORCE GTX 1080 Ti GPUs. We tried both RoBERTa-base and RoBERTa-large. We set hidden size to be 768 for RoBERTa-base and 1024 for RoBERTa-large. We used a batch size of 4 and set max learning rate to be 1e-5. The hidden state dropout percentage is 5%. One epoch takes about 10 minutes. More details can be accessible in our github repo. We got a 73.5% accuracy for the test dataset, which makes us rank 21st out of the 30 teams.

Figure 3: System Performance

Epoch	Accuracy
3	83.21%
6	85.22%
10	87.43%

5 Discussion

There is a big drop in terms of accuracy between training and testing. We doubt it is due to over-fitting. Our system gain very little after 3 epochs so we should stop earlier to make the system more general-purpose. Despite over-fitting, our best accuracy, 87.43% still cannot enter the top 10 on the leader board. We suspect that it is because we did not use any external knowledge base. Some large external knowledge base such as ConceptNet(Speer et al., 2017), could help a lot.

6 Conclusion

We proposed a novel method for fine-tuning pre-trained model and fit it into Subtask B. The combination of InferSent and RoBERTa can make the masked training faster. Meanwhile, (Tamborrino et al., 2020) introduced a more thorough way of masking every word as well as N-grams in both the statement and the explanation. We think that only the important words need mask. Trivial words such as *'he'*, *'the'*, *'to'* could be noise.

Acknowledgements

The author would like to thank the shared task organizers, Shuailong, Cunxiang, etc., for their contribution and organization. The author also wants to thank Mingbo Ma, Hao Tian at Baidu Research, (Bill) Yuchen Lin at USC and Alexandre Tamborrino at Samsung Strategy and Innovation Center for their discussion. Last but not least, big thanks to the anonymous reviewers.

References

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*.

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*.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. *arXiv preprint arXiv:1909.02151*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Alexandre Tamborrino, Nicola Pellicano, Baptiste Pannier, Pascal Voitot, and Louise Naudin. 2020. Pre-training is (almost) all you need: An application to commonsense reasoning. *arXiv preprint arXiv:2004.14074*.

Cunxiang Wang, Shuailong Liang, Yili Jin, Yilong Wang, Xiaodan Zhu, and Yue Zhang. 2020. SemEval-2020 task 4: Commonsense validation and explanation. In *Proceedings of The 14th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.