A study of BERT's processing of negations to determine sentiment

Giorgia Nidia Carranza Tejada[1], Johannes C. Scholtes[1,2], and Gerasimos Spanakis[1][0000−0002−0799−0241]

[1] Maastricht University. Faculty of Science and Engineering, Department of Data Science and Knowledge Engineering
g.carranzatejada@student.maastrichtuniversity.nl,
jerry.spanakis@maastrichtuniversity.nl
[2] iPro Tech LLC-ZyLAB Technologies
j.scholtes@maastrichtuniversity.nl

Abstract. The question considered in this paper is: can BERT effectively distinguish the meaning of the following two sentences: '*BERT is capable of understanding negations*' and '*BERT is not capable of understanding negations*'? This work aims to fulfill the gap in the knowledge about BERT's capacity to handle negations. The specific task under examination is sentiment analysis, where erroneous understanding of negations directly affects the model's performance by wrongly switching polarity of the detected sentiments. In order to determine what BERT 'understands' from negated text, a model was trained and tested by using adversarial conditions. With four distinct configurations, handling negations was studied by interchanging negated sentences during training and testing. The results exposed that in three out of four cases, the BERT's propensity to deal with negations by memorizing information in the large number of connections used by the model, instead of truly understanding the linguistic mechanism of negations. In the remaining case, the model's performance suggested taking decisions based on random features without exposing clear reasoning. Based on these insights, best-practice methods for training BERT to deal better with negations in sentiment analysis can be formulated.

1 Introduction

An area widely investigated in text mining, is sentiment analysis. Sentiment analysis studies techniques to identify and examine human sentiments towards different experiences and interests. In general, the sentiments expressed in a text are positive, negative, or neutral [11].

In order to obtain the correct classification of a sentence, it is essential to handle negations correctly. Not doing so, will impact the polarity of the sentiment, resulting in wrong classifications. An example is the following positive sample: *"this is a good film"*, if it is negated, this sentence expresses a negative opinion: *"this is not a good film"*. So, not dealing correctly with the negation, will change the polarity in the exact opposite direction.

One of the State-of-the-Art (SotA) models to detect and classify sentiments, is BERT (Bidirectional Encoder Representations from Transformer) [14]. Despite the high quality of classification, a detailed error analysis indicates that the majority of misclassifications comes from erroneous handling of negations: a percentage of 66% in comparison to the other error categories. Figure 1 shows the distribution among the classes of errors indicated by [8].

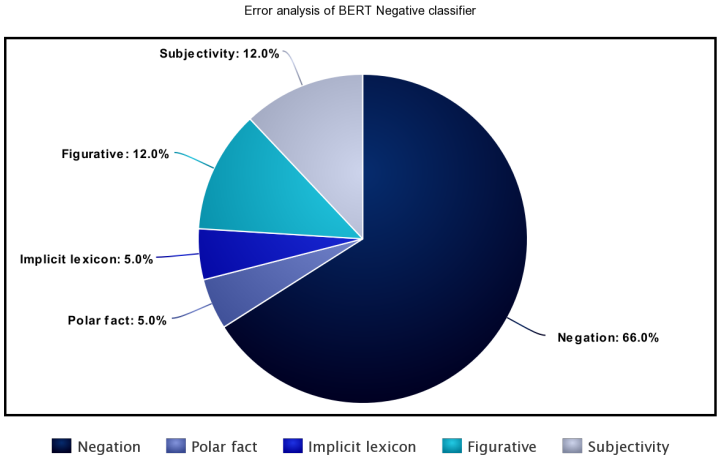


Fig. 1: Distribution of BERT's error analysis for the sentiment analysis task

So, although BERT holds the state-of-the-art result for several Natural Language Processing (NLP) tasks, Figure 1 indicates that BERT is not really capable of handling negations. Given the fact that 66% of errors in the sentiment-analysis task originate from wrongly handling negations, the research in this paper focuses on better understanding BERT's negation-handling mechanisms, in order address these errors and increase the overall result of the sentiment-analysis task.

There are a number of reasons why we believe there is a deeper issue at hand with BERT's negation handling skills: (i) the mechanism behind Word Embeddings assign similar encodings to words used frequently in the same context. In other experiments conucted by [13], it has been observed that words of completely different polarity get very similar encodings (e.g. *good* versus *bad* or *happy* versus *unhappy*). This confuses the classification in tasks such as sentiment analysis, where the polarity is more important than in other tasks such as machine translation. (ii) Due to the enormous amount of trainable connections, BERT has tremendous memory skills. But memorizing is something very different from inferencing the polarity of (double) negations. (iii) BERT's attention mechanism seems to be based on the presence of certain specific cue words it memorizes, thereby missing words relevant for negations such as *not*.

Our work extends prior studies, examining BERT's behavior in an adversarial condition during training and testing. By investigating BERT's predictions based on certain training data, it should be possible to better understand in certain situations where BERT's errors dealing with negations originate from, so they can be better addressed in future sentiment analysis models.

2 Related Work

As stated before, notable advancements in various NLP tasks were produced after the introduction of BERT in 2018. Despite these impressive results, there has also been interest what BERT is not capable of, especially by the computational-linguistic community. For example, [2] analyzed linguistic errors, the problem derived from the commonsense, pragmatic inference, and negation. These experiments showed that BERT failed to adjust to negated statements: the predictions persisted unaltered after the insertion of the negation. See Figure 2 for a number of examples of such wrong prediction. From these, it is completely clear that BERT is completely ignoring the negation!

Context	$\text{BERT}_{\text{LARGE}}$ predictions
A robin is a ____	*bird, robin, person, hunter, pigeon*
A daisy is a ____	*daisy, rose, flower, berry, tree*
A hammer is a ____	*hammer, tool, weapon, nail, device*
A hammer is an ____	*object, instrument, axe, implement, explosive*
A robin is not a ____	*robin, bird, penguin, man, fly*
A daisy is not a ____	*daisy, rose, flower, lily, cherry*
A hammer is not a ____	*hammer, weapon, tool, gun, rock*
A hammer is not an ____	*object, instrument, axe, animal, artifact*

Table 13: $\text{BERT}_{\text{LARGE}}$ top word predictions for selected NEG-136-SIMP sentences

Fig. 2: Table from "[What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models.]", by Ettinger, 2020, *Transactions of the Association for Computational Linguistic*

This work was a direct extension of [3], which focused on analyzing BERT's syntactic abilities by supplying an entire sentence to BERT, while masking out the single focus verb.

Another study proposed by [7] proved the deterioration of BERT performance when denials were added in claims for argument comprehension tasks.

[5] investigated the effect of the negation on the question-answering task by applying the masked language model. The research concluded that BERT's can learn predictions based on exact phrases shown during training, whereas it

poorly generalizes over a test set that contains phrases it did not see during training.

Also, [4] focused on understanding how a Pre-Trained Language model (hereinafter PLM) like BERT learns factual knowledge from the training data. During the symbolic reasoning analysis, e.g. the ability of a PLM to deduce information not shown during the pre-training, a rule explicitly investigated was negation. The work assumed that the general concept of denial was not understood while co-occurrence is used to acquire antonym negation. Kassner's study involved [9]'s prior work, which connects BERT's prediction to the knowledge-base and (lack of actual) inferencing capabilities.

Other studies into the relation of BERT and negation handling can be found in [6], which focused on employing BERT to detect the denial and delimiting its scope, and [15], which analyzed a plausible relation between a negation cue and its scope in the attention heads. Both confirmed lack of actual knowledge of the negations by measuring significant inconsistencies in the average negation detection.

Our approach is similar to the work of Ettinger and Goldberg, as we focus more on BERT's word prediction capability, specifically on the sentiment analysis task. Our work differs from Kassner's, which focuses more on the detection of negations while we examine the effect of negations on sentiment prediction.

3 Methodology

BERT's strength comes from the simplicity of adjusting the original pre-trained model configuration for a specific task by fine-tuning the model, thereby taking advantage of transfer learning. Instead of having to learn language from scratch, BERT has basic linguistic skills resulting from being exposed to many billions of words in their linguistic context. But, for reasons not well understood, this ability does not apply to negation handling, which obviously is very important in sentiment analysis. The question we ask ourselves: is BERT just memorizing training data using its large number of parameters, or does it actually "understands" negations?

From initial observations, our hypothesis is that BERT tends to memorize. To verify the validity of this assumption, BERT's ability to deal with negations will be studied in adverse training and testing conditions. The examination will follow a similar approach as Ettinger's work (see figure 2 for examples), where the knowledge of BERT is questioned by adding negations to testing sentences that the model did not see before. So, if the model is indeed only memorizing, the prediction will remain unchanged after these perturbations.

To begin with, two BERT classifiers are trained with labeled sentences from the SemEval 2017 task 4a ([11]) and SST5 ([12]) data set. One binary positive classifier and one binary negative classifier ([1]). Subsequently, classification is tested on negated sentences for each class. From the examples in figure 3, it can be observed that in both cases BERT predicts the original sentiment and not the negated sentiment.

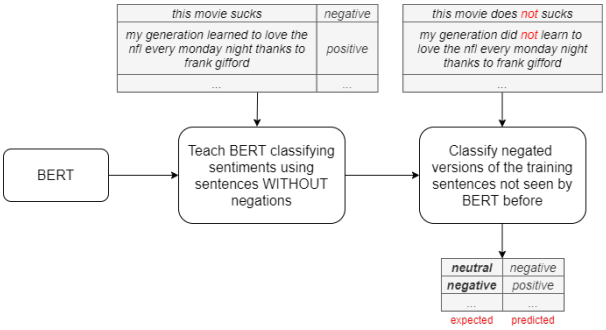


Fig. 3: BERT's misclassifications after the addition of negations.

In subsequent experiments, four different approaches were used to confirm that BERT actually memorizes. First, in model M1 we will include only sentences without negations in the training set. The test set then contains negated versions of these sentences. In model M2 , we take the reverse approach, where the training set only includes negated sentences and the test set then contains non-negated versions of these sentences. Then, model M3 will include in the training data set randomly one of the two versions of the sentence. The inverted sentence of each will be considered in the testing. So, in M3 the system will be exposed to both negated and non-negated sentences, but in the testing sentences, the negations will be opposite from the training sentences. Finally, model M4. includes for every sentence both the negated and the non-negated version. We then test on a validation set, containing sentences not seen before by the model during training.

The configurations selected for the training and testing are briefly explained in table 1

Model	Negated	Not Negated
M1	none	all samples
M2	all samples	none
M3	either	either
M4	all samples	all samples

(a) The data set employed during the training.

Model	Negated	Not Negated
M1	all samples	none
M2	none	all samples
M3	remaining	remaining
M4	test samples	test samples

(b) The data set employed for the testing of the model.

Table 1: The tables represent the configuration examined to verify the performance of BERT towards the negations.

If experiment M1 and M2 result in BERT predicting the original sentiment from training instead of the negated sentences, and if BERT shows inconsistent

behavior for experiment M3, then it is clear that BERT is actually memorizing on cue words instead of really understanding the linguistic operation of negation.
In order to demonstrate this more convincingly, the behavior of connotations such as *can not* versus *cannot* versus *can't* will be investigated. Because, if BERT can deal with *cannot* and *can't* but not with *can not*, then the case for memorization of cue words is even stronger.
This, and the influence of specific negations words such as *not* is studied by using through the Local Interpretable Model-Agnostic Explanations (hereinafter LIME) approach. LIME is a technique employed to explain a prediction of any black-box machine learning model by presenting qualitative connections between the instance's components and the model's prediction [10]. The methodology proposed by LIME consisted on performed a local fidelity analysis by initially altering the original data point before being fed into the model. Then, the importance of each feature is represented by the change in the predictions obtained.

The interpretation obtained is not the faithful representation of the entire model but is reliable locally, which depends on the performance obtained in the proximity of the sample examined. Additionally, LIME guarantees an interpretable representation by applying bag-of-word when it is needed for text classification.

In the example proposed by the paper, the technique helps to understand the eventual cue words learned by the model to determine the class of the text. The explanations evidenced an issue of the classifier related to the data set selected. The same approach will be used during this examination to provide more insights into the impact of negation words on the sentiment predictions.

4 Experiments

For the experiments, a sentiment classifier was built using the Transformers library by HuggingFace supported by the PyTorch Machine Learning framework. The number of epochs employed in our experiment is equal to 2, the number of batches is set to 32, and the optimizer selected was AdamW. The model loaded from the Transformers library, represented only the hidden layer of the input tokens. The output is passed through linear transformation.

Since the main purpose of the baseline([1]) was to detect either positive or negative sentiment in a neutral context, it was considered to employ a binary classification using the one-vs-rest technique. The choice derived from the over-representation of the neutral class influenced the multi-class model performance to assign an incorrect neutral label in most cases. So, the sentiment analysis system consists of two binary classifiers: the positive classifier trained as positive against either negative or neutral and the negative classifier trained as negative against either positive or neutral.

To sum up, each sentence to be evaluated is fed into both the classifiers as input. Then, the outputs, which correspond to the outcomes of the binary classifiers, identify the sentiment in the text.

Starting data sets for our analysis are the concatenation of SemEval-2017 and SST5, as those used for the baseline model. The first was provided by task 4a of the International Workshop on Semantic Evaluation (SemEval) 2017, Sentiment Analysis in Twitter1. CrowdFlower or Mechanical Turk realized the annotations for each tweet [11]. Besides, the labels complied with the three sentiment categories previously nominated, and they are distributed as follows: 34% positive, 16% negative, and 50% neutral.

Then, the data set SST5, published on [12], is a fine-grained sentiment data set containing five different labels: 0 (very negative), 1 (negative), 2 (neutral), 3 (positive), 4 (very positive). Because our experiments consider only three labels (negative, neutral, and positive), the labels *very positive* and *very negative* were included in respectively *positive* and *negative*. The overall balance of the labels in our combined data set is defined as follows: 42% positive, 39% negative, and 19% neutral.

Name	Size	Positive	Negative	Neutral
SemEval 2017 task 4a	20631	34%	16%	50%
SST5	8544	42%	39%	19%

Table 2: The size and labels' distribution of the sentiment data sets.

The original data set presented an unbalance among the two categories: negated cases were only 22% of the entire collection. To balance, the data set was augmented through transformation functions as defined in table 3. This then resulted in a more balanced distribution, with 49.4% negated sentences and 50.6% not negated ones, and in different data sets for experiments M1, M2, M3. and M4. For training 90% of the modified data is used. For testing the remaining 10%.

Transformation function	**Original sentence**	**Modified sentence**	**Created samples**
Addition of the negation	Paul Bettany is cool	Paul Bettany is *not* cool	15792
Removal of the negation	he 's *not* good with people	he 's good with people 1	8569

Table 3: The table defined the transformation functions used in the examination to alter the original data.

5 Results and Discussion

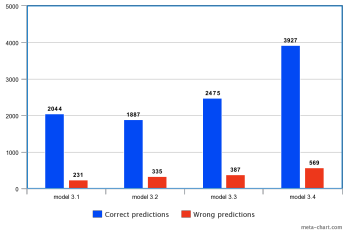
Table 4 collects the outcomes of the binary classifiers obtained from the test on the original sentences, which are the samples following the configurations established for the training data. So this table represents how well the classifier works on similar sentences to the training data. While table 5 assembles the performances on negated versions these sentences.

	Positive			Negative		
Model	**Precision**	**Recall**	**F_measure**	**Precision**	**Recall**	**F_measure**
M1	0.75	0.79	0.77	0.73	0.76	0.74
M2	0.93	0.96	0.94	0.87	0.94	0.90
M3	0.90	0.92	0.91	0.82	0.84	0.83
M4	0.88	0.90	0.89	0.84	0.89	0.86

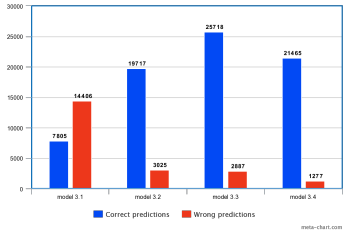
Table 4: Original sentences

	Positive			Negative		
Model	**Precision**	**Recall**	**F_measure**	**Precision**	**Recall**	**F_measure**
M1	0.89	0.28	0.42	0.75	0.20	0.32
M2	0.51	0.93	0.66	0.66	0.63	0.65
M3	0.83	0.86	0.85	0.79	0.79	0.79
M4	0.88	0.89	0.89	0.84	0.89	0.86

Table 5: Negated sentences

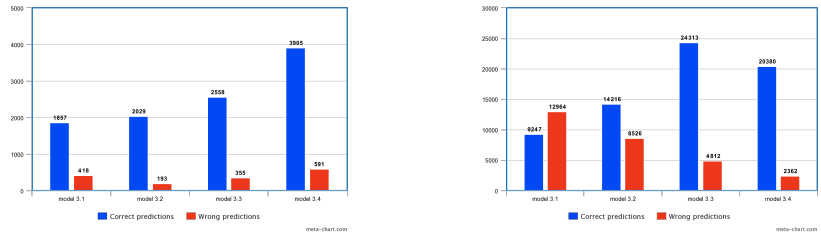


(a) Original sentences



(b) Negated sentences

Fig. 4: Distribution of the correct and wrong predictions for the **negative** classifier (model M1 to M4)



(a) Original sentences (b) Negated sentences

Fig. 5: Distribution of the correct and wrong predictions for the **positive** classifier (model M1 to M4)

5.1 Model M1, trained with non-negated sentences and tested on negated versions of these sentences.

Model M1 was trained by not showing negations during training. Then, to question BERT's capabilities to deal with negations, training sentences were negated and used as test.

As expected, BERT is not able to handle the negations in the test sentences. Indeed, the performance drops significantly on the negated sentences compared to the non-negated ones, as can be observed from the high error in the left columns of figure 4b and 5b.
Furthermore, LIME was used to examine the wrongly predicted sentence: '*about to go shopping again tomorrow bc the dress I got for jason aldean is not cute.*' in more detail.

The respectively original sentence was: '*about to go shopping again tomorrow bc the dress I got for jason aldean is cute*', and holds a positive sentiment. When negated, the prediction should be reversed and classified as negative, but the negative classifier failed to identify the sentiment. Figure 6 provides more insight how the prediction was made by using LIME. The LIME's process to represent the feature's impact in the prediction evidenced that the negation cue and its scope had been recognized by BERT and correctly attributed to the negative class. However, the significance of these words was not enough to invert the overall label since the distance of the class none was still considerable compared to the negative class, which was the correct prediction.

5.2 Model M2, trained with negated sentences and tested on non-negated versions of these sentences.

Model M1 is the reverse of model M2: training included only negated sentences, where testing was done with non-negated versions of the training sentences.

In this case, the performance in the non-negated sentences did not decrease as drastically as in model M1. Additionally, it was observed that the most common error originated from issues dealing with subjectivity/objectivity and figurative

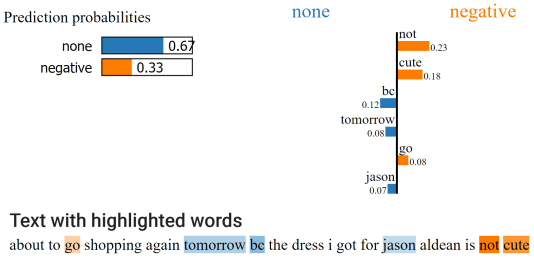


Fig. 6: The result of LIME in the sentence '*about to go shopping again tomorrow bc the dress I got for jason aldean is not cute.*' for the negative classifier.

language. Therefore, the result seems to indicate that BERT was actually learning something about dealing with negations.
However, further analyzing the model in more details with respect to connotations such as *cannot* or *can't* versus *can not*, it becomes clear that BERT's decisions are actually still based on memorization. Considering that the training data included one version of a negated verb, either extended (*is not*) or contracted (*isn't*), then the prediction should not be influenced if the negated verb is replaced by the contracted connotation in the test data. For example, table 6 represents the situation including the same sentence: firstly with the verb negated by using a separate *is not* and then by using the contracted connotation: *isn't*. The prediction of the system should be equal, but in our case, it changed for the contracted connotation. So, evidently, BERT did not "understand" negations but based it's decision on memorization.

text	label	prediction
solondz is so intent on hammering home his message that he does forget to make it entertaining	negative	-
solondz is not so intent on hammering home his message that he does not forget to make it entertaining	positive	negative
solondz isn't so intent on hammering home his message that he doesn't forget to make it entertaining	positive	positive

Table 6: An example of the classification of the same sentence but changing the type of negated verb: firstly the extended, then the connotation.

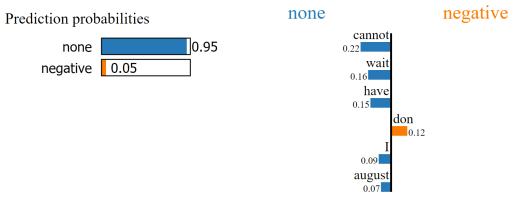
5.3 Model M3, Trained on either negated or non-negated sentences and tested on the inverse for each sentence.

The third model examined was M3, which was trained on the data set composed of a random selection of one version of a sentence: either the negated or the non-negated version. Then, testing was done on the inverse of the train sentences.

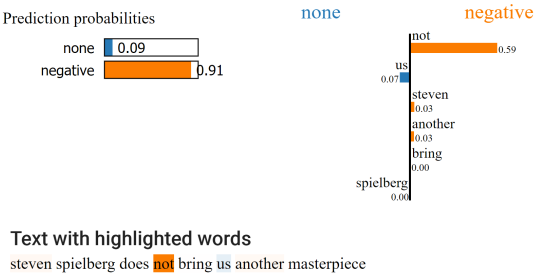
So, for the negated sentences, a non-negated one was used and visa-versa. In this case, there were more correctly predicted sentences than in model M1 and M2.

Even though, by examining in-depth the correctly or wrongly classified sentences, it was not possible to deduct any particular pattern why some sentences were classified correctly or wrongly. Consequently, the model seemed to take decisions based on random features, which were not clearly understandable from either the predictions or through the deployment of LIME.

In particular, the examination aimed to identify a common pattern, similar to the previous case, where the negation's impact was repeatedly neglected for the prediction, or the use of connotations underlined an inaccurate behavior of the model. In this state, no anomalies were detected from the employment of connotations, and as well, the negations were sometimes correctly classified and sometimes not, without establishing a frequent behavior. For instance, the latter case was represented in the results obtained by LIME in two sentences. First, an exact classification correctly identified and handled the negation. Then, an erroneous prediction was determined from a sentence with a double negation. The initial negation did not alter the sentence, but the second did by reversing it. Although the second negation cue was identified and attributed to a negative class, the impact on the total prediction was not decisive.



(a) Wrongly predicted sentence.



(b) Correct predicted sentence.

Fig. 7: LIME results on one correct and one wrong prediction of model M3.

5.4 Model M4, Train with both negated and non-negated versions of each sentence, and test on an external validations set

Finally, the last model, M4, obtained the highest result in the testing on negated sentences. Additionally, it achieved the smaller variation in the precision, recall, and f-measure between the original and negated results. This performance was strictly connected to the configuration adopted by the model since the testing was on a direct subset of the training data. Therefore, from this last experiment, one could derive that BERT capabilities to deal with negations are based on memorization.

6 Conclusion

The goal of this research was to contribute to a better understanding of the underlying mechanisms of BERT's negation handling. Therefore BERT's behavior was investigated by testing adversarial sentences in sentiment analysis, where the effects of wrong negation handling has much more impact than in other linguistic tasks.

This research indicates that BERT's handling of negations is more based on it's tremendous ability to memorize rather than "understanding" the negation.

Notably, BERT was unable to learn how to properly deal with denial when trained only on denied or non-denied sentences. Indeed, in the first case, the model's predictions turned out to be random, as evidenced by the connotation example, while in the second case, the model ignored the negation. The optimum results were achieved by the last model, where each sample included both the negated and non-negated version of all sentences. This configuration was able to take full advantage of BERT's memorization capabilities and resulted in the highest f1 scores.

In conclusion, for future realization of sentiment analysis systems where negations will be properly addressed, it is recommended that the data sets contain a proportioned distribution of negated and non-negated cases for each sentence. Additionally, it is also suggested that future data sets for sentiment analysis competitions (e.g. the ones used in SemEval and SST5), spend more attention to dealing with negations, as this is a major source of errors in the real-world.

7 Acknowledgements

The authors wish to thank ZyLAB Technologies BV in Amsterdam, the Netherlands for providing the funding and resources to realize this research. ZyLAB's willingness to allow a data science team to investigate the applications of various new methods and technologies in the field of text-mining is very much appreciated.

References

1. Carranza Tejada, G.N.: A study of word embeddings and support vector machines for emotions and sentiments recognition. Tech. rep. (2020)
2. Ettinger, A.: What bert is not: Lessons from a new suite of psycholinguistic diagnostics for language models. Transactions of the Association for Computational Linguistics **8**, 34–48 (2020)
3. Goldberg, Y.: Assessing bert's syntactic abilities. arXiv preprint arXiv:1901.05287 (2019)
4. Kassner, N., Kroje, B., Schütze, H.: Pre-trained language models as symbolic reasoners over knowledge? arXiv preprint arXiv:2006.10413 (2020)
5. Kassner, N., Schütze, H.: Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. arXiv preprint arXiv:1911.03343 (2019)
6. Khandelwal, A., Sawant, S.: Negbert: A transfer learning approach for negation detection and scope resolution. arXiv preprint arXiv:1911.04211 (2019)
7. Niven, T., Kao, H.Y.: Probing neural network comprehension of natural language arguments. arXiv preprint arXiv:1907.07355 (2019)
8. Novielli, N., Girardi, D., Lanubile, F.: A benchmark study on sentiment analysis for software engineering research. In: 2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR). pp. 364–375. IEEE (2018)
9. Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A.H., Riedel, S.: Language models as knowledge bases? arXiv preprint arXiv:1909.01066 (2019)
10. Ribeiro, M., Singh, S., Guestrin, C.: " why should i trust you?" explaining the pre-dictions of any classifier. Proceedings of the 22ndACM SIGKDD international conference on knowledge discovery and data mining p. 1135–1144 (August 2016)
11. Rosenthal, S., Farra, N., Nakov, P.: SemEval-2017 task 4: Sentiment analysis in Twitter. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017). pp. 502–518. Association for Computational Linguistics, Vancouver, Canada (Aug 2017). https://doi.org/10.18653/v1/S17-2088, `https://www.aclweb.org/anthology/S17-2088`
12. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A.Y., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 conference on empirical methods in natural language processing. pp. 1631–1642 (2013)
13. Tang, D., Wei, F., Qin, B., Yang, N., Liu, T., Zhou, M.: Sentiment embeddings with applications to sentiment analysis. IEEE transactions on knowledge and data Engineering **28**(2), 496–509 (2015)
14. Yadollahi, A., Shahraki, A.G., Zaiane, O.R.: Current state of text sentiment analysis from opinion to emotion mining. ACM Computing Surveys (CSUR) **50**(2), 1–33 (2017)
15. Zhao, Y., Bethard, S.: How does bert's attention change when you fine-tune? an analysis methodology and a case study in negation scope. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 4729–4747 (2020)