Semantic similarity prediction is better than other semantic similarity measures

Anonymous authors
Paper under double-blind review

Abstract

Semantic similarity between natural language texts is typically measured either by looking at the overlap between subsequences (e.g., BLEU) or by using embeddings (e.g., BERTScore, S-BERT). Within this paper, we argue that when we are only interested in measuring the semantic similarity, it is better to directly predict the similarity using a fine-tuned model for such a task. Using a fine-tuned model for the STS-B from the GLUE benchmark, we define the STSScore approach and show that the resulting similarity is better aligned with our expectations on a robust semantic similarity measure than other approaches.

1 Introduction

When we are considering research questions related to the quality of the output of generative models for natural language (e.g., Large Language Models/LLMs like GPT(-2) by Radford et al. (2019) and all the other currently very successful models that followed), we need to determine if the generated text is a valid output for the desired task. For example, when we want to use such a model for translation from English to German, we need to determine if the resulting German sentence has the same meaning as the English sentence. Since there are typically many possible valid outputs for such tasks, we cannot easily specify a ground truth benchmarks. While an expert would be able to determine if the output is correct, automating this task with a computer is not easy, as this would effectively require solving the translation problem first. However, since scaling such evaluations to thousands or even millions of examples is not possible without automation, researchers created different methods to assess the quality of output using the idea of the *semantic similarity*.

The general idea is quite simple: while we cannot state the ground truth, we can state the semantic meaning of one or more correct outputs. Simple metrics such as the Word Error Rate (WER) estimate this through the consideration of the differences between words, similar to the Levenshtein distance. Others, like BiLinigual Evaluation Understudy (BLEU) (Papineni et al., 2002) or Recall-Oriented-Understudy for Gisting Evaluation (ROGUE) (Lin, 2004) are rather looking at the overlap between subsequences of texts, assuming that a larger overlap means a better semantic similarity. Furthermore, some methods (e.g., BLEU) go beyond the comparison of single solutions to a sample output and instead allow the comparison between corpora of generated and sample solutions. For individual words without considering them in their context, word embeddings like word2vec (Mikolov et al., 2013) in combination with cosine similarity are a generally accepted way to estimate the semantic similarity between words, even though this approach has problems with ambiguities, e.g., caused by polysemes (Del Tredici & Bel, 2015). Similar methods were extended to whole sentences (Sharma et al., 2017), but they did not achieve the same level or performance.

The transformer architecture enabled the context-sensitive calculation of embeddings (Vaswani et al., 2017). Naturally, this was adopted for the calculation of the similarity. Two methods based on this are currently primarily used. The first is BERTScore by Zhang et al. (2020), which is based on an optimal matching of the pair-wise similarities of words within a contextual BERT embedding and has been used by thousands of publications since its publications in 2020. The other is Sentence-BERT (S-BERT) by Reimers & Gurevych (2019), who pool the embeddings of the tokens into an embedding for sentences. The cosine similarity can

then be computed between these sentence embeddings, same as between words with word2vec or with earlier, less powerful, sentence embeddings (e.g. Sharma et al., 2017).

This success not withstanding, we want to argue that this approach should not be further pursued in favor of a simpler solution to estimate the semantic similarity, i.e., simply predicting the semantic similarity with a regression model. We note that this idea is not new and was, e.g., also used to define approaches like BEER (Stanojević & Sima'an, 2014) or RUSE (Shimanaka et al., 2018). However, these models are from the pre-transformer era of natural language processing. As can be seen in large benchmarks like GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019), models based on the transformer architecture (Vaswani et al., 2017) provide a lot better performance. Therefore, we formulate the following hypothesis for our research:

Hypothesis: Modern language models with an encoder-only transformer architecture similar to BERT (Devlin et al., 2019) that are fine-tuned as regression models for the similarity between sentence pairs are also capable to robustly measure the semantic similarity beyond their training data and are better measures for the semantic similarity than embedding-based and n-gram approaches.

We derive this hypothesis from the assumption, that if such a regression model fulfills its task, employing it as a semantic similarity measure would be the natural use case beyond just benchmarking model capabilities. Within this paper, we present the results of a confirmatory study, that demonstrates that downloading a fine-tuned RoBERTa model (Liu et al., 2019) for the STS-B task (Cer et al., 2017) from the GLUE benchmark from Huggingface and using this model to predict the similarity of sentences fulfills our expectations on a robust similarity measure better than the other models we consider. We refer to this approach as STSScorer. To demonstrate this empirically, we compute the similarity score for similarity related GLUE tasks and show that while the predictions with the STSScorer are not perfect, the distribution of the predicted scores is closer to what we would expect given the task description, than for the other measures.

2 Method

Within this section, we describe the research method we used to evaluate our hypothesis. We first describe the STSScorer model for the prediction of the similarity in Section 2.1. Then, we proceed to describe our analysis approach in Section 3, including the expectations we have regarding our hypothesis and the tools we used in Section 3.1.

2.1 STSScorer

Listing 1 describes our approach: we download a fine-tuned model from Huggingface for the STS-B task (Held, 2022), which was based on RoBERTa (Liu et al., 2019). The STS-B tasks contains sentence pairs from news, image captions and Web forums. Each sentence pair received a score between zero and five. These scores were computed as the average of the semantic similarity rating conducted by three humans such that five means that the raters believe the sentences mean exactly the same and zero means that the sentences are completely unrelated to each other. We simply use the model trained for this task and divide the results by five to scale them to the interval $[0, 1]$. Hereafter, we refer to this model as STSScorer.

3 Analysis approach

Within this section, we describe the research method we used to evaluate our hypothesis. Our approach is similar to the method used by Zhang et al. (2020) for the evaluation of the robustness of similarity measures also by Reimers & Gurevych (2019) for the evaluation of S-BERT: we utilize data labeled data for which we a have an expectation of what to observe, when measuring the semantic similarity.

There are three such data sets within the GLUE benchmark:

- the Semantic Textual Similarity Benchmark (STS-B) data we already discussed above;

Listing 1: A simple class to define a fully functional semantic similarity scorer based on a pre-trained model for the STS-B tasks.

```
import transformers

class STSScorer:
  def __init__(self):
    model_name = 'WillHeld/roberta-base-stsb'
    self._sts_tokenizer = transformers.AutoTokenizer.from_pretrained(
        ↪ model_name)
    self._sts_model = transformers.AutoModelForSequenceClassification.
        ↪ from_pretrained(model_name)
    self._sts_model.eval()

  def score(self, sentence1, sentence2):
    sts_tokenizer_output = self._sts_tokenizer(sentence1, sentence2, padding=
        ↪ True, truncation=True, return_tensors="pt")
    sts_model_output = self._sts_model(**sts_tokenizer_output)
    # logits contain regression values
    # need to divide by five due to scoring approach of STS-B between 0 and 5
    return sts_model_output['logits'].item()/5
```

- the Microsoft Research Paraphrase Corpus (MRPC, Dolan & Brockett (2005)) data, where the task is to determine if two sentences are paraphrases; and
- the Quora Question Pairs (QQP, Iyer et al. (2017)) data, where the task is to determine if two questions are duplicates.

For STS-B and MRPC, we use the test data. Since the labels for QQP's test data are not shared, we use the training data instead. To the best of our knowledge this data was not seen during the training of the STSScorer and and S-BERT models of the models we use, as Quora was not part of the pre-training corpus of RoBERTa, which mitigates the associated risks regarding data contamination. However, model underlying S-BERT was fine-tuned using constrastive learning on a corpus of one billion sentences (Reimers & Gurevych, 2019), which contained about 100,000 instances from the QQP data, i.e., about a quarter. Thus, S-BERT might have an advantage on this data.

On each of these data sets, we compute the similarity between all pairs of sentences with BLEU, BERTScore, S-BERT,[1] and STSScore. All methods we consider compute scores between zero (not similar at all) and one (same semantic meaning), which simplifies the direct comparison. This narrower view of few models allows us to consider the results more in-depth. Specifically, we can go beyond the plain reporting of numbers, and instead look directly at the distributions of the similarity measures for different data sets. Due to the confirmatory design of our study, we formulate concrete expectations on the results, given the properties for each data set. How well the similarity measures fulfill these expectations will be later used to evaluate our hypothesis. Based on the strong performance of BERT-based models on the STS-B task in the GLUE benchmark, we predict based on our hypothesis that STSScorer should have the best alignment with our expectations for all data sets.

We note that while we could have added more approaches to this comparison, e.g., ROUGE (Lin, 2004), METEOR (Lavie & Agarwal, 2007), RUSE (Shimanaka et al., 2018), and BEER (Stanojević & Sima'an, 2014), these models were all already compared to BERTScore, which was determined to provide a better measure for the similarity (Zhang et al., 2020). Further, we refer to a general overview on such metrics to the work by Zhang et al. (2020). Thus, instead of providing a broad comparison with many models, we rather compare our approach to the embedding-based aporoaches S-BERT and BERTScore which are currently

[1] From now on, we simply refer to calculating the cosine similarity between embeddings with S-BERT as S-BERT for brevity.

used as de-facto state-of-the-art by most publications and the still very popular BLEU method that uses an n-gram matching approach.

3.0.1 STS-B data

While the comparison on the STS-B data may seem unfair, because the STSScorer was specifically trained for that model, the analysis of the behavior of the different scorers on this model still gives us interesting insights: for any semantic similarity measure, the distribution of the scores should be directly related to the label of STS-B,[2] which is a human judgement of the semantic similarity. Consequently, when we plot the label on the x-axis versus a semantic similarity score on the y-axis, we would ideally observe a strong linear correlation. Visually, we would observe this by the data being close to the diagonal. A less ideal, but still good, result would be that the trend is monotonously increasing indicating a rank correlation, which would mean that while the magnitudes of the similarity measure are not aligned with the human judgements from STS-B, at least the order of values is. Any other trend would mean that the similarity measure is not aligned with the human judgements of STS-B. In addition to this visualization we also measure linear correlation between the scores and the labels with Pearson's r and the rank correlation with Spearman's ρ, as is common for the STS-B task within the GLUE benchmark Wang et al. (2018) and was also used by Reimers & Gurevych (2019) for the evaluation of S-BERT.

Because the STSScorer was fine-tuned on the STS-B data, we only utilize the test data. Nevertheless, because this is exactly the same context as during the training of STSScorers models (same data curators, same humans creating the judgements) this model has a huge advantage over the other approaches. Due to this, we expect that STSScorer is well aligned with the human judgements and we observe the linear trend described above. If this fails, this would directly counter our hypothesis, as this would not even work within-context. The BLEU, BERTScore, and S-BERT models were created independent of the STS-B data, but given their purpose to estimate the semantic similarity, they should still be able to fulfill the desired properties. If this is not the case, this would rather be an indication that these models are not measuring the semantic similarity – at least not according to the human judgements from the STS-B data.

3.0.2 MRPC and QQP data

The MRPC and QQP data are similar: both provide binary classification problems. With MRPC, the problem is paraphrasing. With QQP the problem is duplicate questions, which can also be viewed as a type of paraphrasing, i.e., the paraphrasing of questions. Paraphrasing is directly related to semantic similarity, as paraphrased sentences should be semantically equal. Thus, similarity measures should yield high values for paraphrases and duplicate questions, ideally close to one.

For the negative examples of MRPC, a look at the data helps to guide our expectations. When considering the MRPC data, we observe that the negative samples are all sentences on the same topic, with a different meaning. As example, consider the first negative example from the training data:

Sentence 1: *Yucaipa owned Dominick 's before selling the chain to Safeway in 1998 for $ 2.5 billion*

Sentence 2: *Yucaipa bought Dominick 's in 1995 for $ 693 million and sold it to Safeway for $ 1.8 billion in 1998 .*

Both sentences are related to the ownership of *Dominick's* by *Yucaipa* and the sale to *Safeway*, but consider different aspects regarding the time and different amounts of money for the payment. Thus, we observe semantic relationship, but not a paraphrasing. While we have not read through all negative examples from the MRPC data, we also did not find any examples where both sentences were completely unrelated. Consequently, we expect values for the semantic similarity of the negative examples to be significantly larger than zero, but also smaller than of the positive examples of actual paraphrases with a notable gap.

[2]While STS-B is a regression task and and it would be better to speak of a dependent variable here, we rather speak of labels all the time to be consistent with the subsequent classification tasks.

	STS-B		MRPC		QQP	
	r	ρ	*neg*	*pos*	*neg*	*pos*
BLEU	0.34	0.32	0.26 (0.19)	0.39 (0.20)	0.11 (0.23)	0.18 (0.25)
BERTScore	0.53	0.53	0.53 (0.15)	0.68 (0.13)	0.44 (0.26)	0.67 (0.17)
S-BERT	0.83	0.82	0.71 (0.15)	0.83 (0.12)	0.56 (0.23)	0.86 (0.10)
STSScore	0.90	0.89	0.61 (0.18)	0.84 (0.13)	0.44 (0.24)	0.76 (0.18)

Table 1: Summary statistics of the results. Pearson's r and Spearman's ρ between the labels and similarities for STS-B. Mean values with standard deviation in brackets for both classes of the MRPC and QQP data. We use *neg*/*pos* to indicate the classes such that *pos* is the semantically equal class. All values are rounded to the second digit.

For the negative examples of QQP, the expectation is less clear. For most pairs, we observe that both questions are somewhat related, i.e., different questions regarding the same topic. As example, consider this negative example from the training data:

Sentence 1: *What causes stool color to change to yellow?*

Sentence 2: *What can cause stool to come out as little balls?*

Both questions are about *stool*, but different aspect of stool. This kind of difference is similar to what we have within the MRPC data. However, we also observed examples, were the questions are completely unrelated. Consider the following instance from the training data of QQP:

Sentence 1: *How not to feel guilty since I am Muslim and I'm conscious we won't have sex together?*

Sentence 2: *I don't beleive I am bulimic, but I force throw up atleast once a day after I eat something and feel guilty. Should I tell somebody, and if so who?*

While both questions are broadly related to the concept of *guilt*, the rest is completely unrelated and we would expect a very low semantic similarity. Consequently, while our expectation for the semantic similarity measures for the majority of the negative samples is similar to that of MRPC (i.e., significantly larger than zero, but smaller than for the positive examples), we also expect to observe a strong tail in the distribution with lower similarities.

For both data sets, we visualize the distribution of the similarities per class. Additionally, we report the central tendency (arithmetic mean) and variability (standard deviation) per class in the data.

3.1 Tools used

We used the Huggingface transformer library to implement STSScore and the Huggingface evaluation library for BLEU. For S-BERT, we used the `all-MiniLM-L6-v2` which was tuned for high-quality sentence embeddings using constrative learning model (Reimers, 2021; Reimers & Gurevych, 2019) and the python package provided by Reimers & Gurevych (2019). For BERTScore we used the default RoBERTa model for the English language and the python package provided by Zhang et al. (2020). We used Seaborn for all visualizations and Pandas for the computation of the correlation coefficients, mean values, and standard deviations. All implementations we created for this work are publicly available online: `REMOVED`

4 Results

Figure 1 shows the results on the STS-B test data. The STSScore has the expected strong linear correlation with the labels. However, this is not surprising, since the underlying model was fine-tuned for this task and the strong performance was already reported through the GLUE benchmark. Still, this confirms the semantic similarity prediction fulfills the expectation we have on a semantic similarity measure. S-BERT is

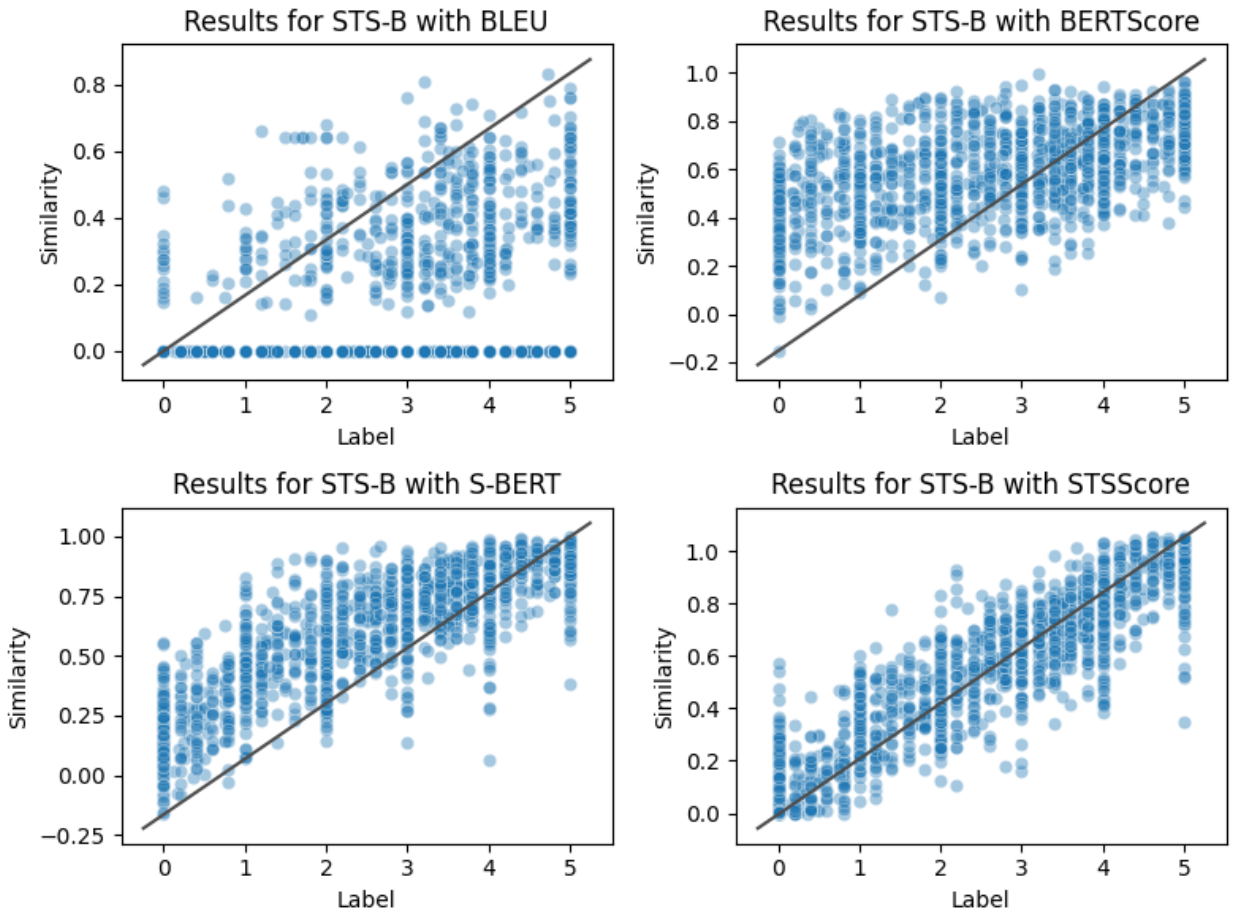

Figure 1: Evaluation of similarity measures on the test data of STS-B. Ideally, the similarity correlates linearly with the labels, i.e., the scores are close to the black line.

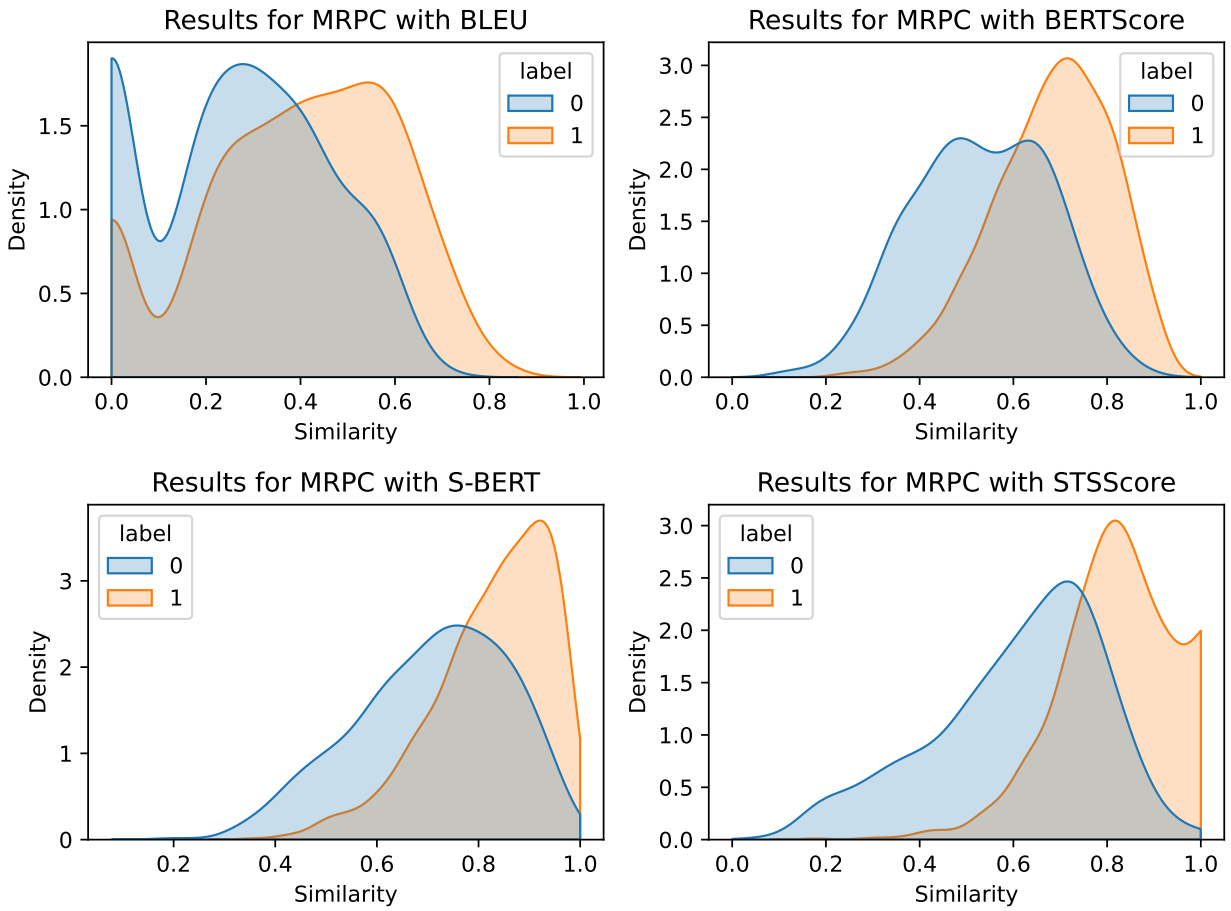

Figure 2: Evaluation of similarity measures on the test data of MRPC. Ideally, the positive class (1) has scores close to one and the negative class (0) has smaller values, but not close to zero.

also seems to have the desired linear correlation, but with a general tendency to overestimate the similarity as most values are above the diagonal. The same cannot be said about BLEU or BERTScore. Both are not aligned at all with the expectations from the STS-B task. The values of BERTScore rather seem fairly randomly distributed, the values of BLEU are often exactly zero and otherwise often a lot lower than expected. An optimistic reading of the BERTScore results detects a weak upward slope in the similarity scores that would be expected. The correlation coefficients depicted in Table 1 match the results of our visual analysis: STSScore is strongly correlated ($r = 0.90$, $\rho = 0.89$), followed by S-BERT that is also strongly correlated, but generally a bit weaker than STSScore ($r = 0.83$, $\rho = 0.82$). BERTScore has only a moderate correlation ($r = 0.53$, $\rho = 0.53$), the correlation of BLEU is weak ($r = 0.34$, $\rho = 0.32$).

Figure 2 shows the results for the MRPC data, Table 1 shows the statistical markers. STSScore yields the expected results: the paraphrasings have a higher similarity than the negative examples, with typically high scores (mean=0.84). However, the density plot shows that the scores are not always close to one, though only few scores are below 0.6. We also observe that the non-paraphrasings are almost always detected semantically somewhat similar (mean 0.61) with a high variability that covers nearly the complete range. However, we note that the density drops to almost zero at very high values (>0.9) and very low values (<0.1). This is in-line with our expectations: the similarity measure typically does not indicate unwarranted equality and it picks up the relationships within the data not dropping to zero. S-BERT is generally similar with high scores for the paraphrasings (mean=0.83). The distribution looks a bit different from STSScore: while STSScore as a peak at around 0.82 and a second peak at exactly one, S-BERT has only a single peak at about 0.92, but drops sharply after this peak. The non-paraphrasings have a higher semantic similarity

than for STSScore (mean=0.71), which aligns with the tendency of S-BERT to overestimate the similarity that we also observed with the STS-B data.

The results of BERTScore exhibit a lot of the expected properties, i.e., a larger mean value for the paraphrases and the similarity for the negative examples covers the whole range, except for the very high and very low values. However, we note that there are only few cases with a similarity close to one for the paraphrases, even though this is actually the expected value. Moreover, the peak of the distribution is also a bit lower than for STSScore and the tail towards lower values for paraphrases is also stronger. As a result, the mean value for the paraphrases of BERTScore is only 0.68 for the positive examples. Moreover, these downward shifts in distribution also lead to a larger overlap between the distributions for the paraphrases and the negative examples, i.e., there is a stronger random element in observing large scores with BERTScore than with STSScore.

For BLEU, the results are not fully aligned with our expectations for the MRPC data. BLEU displays the same tendency for often reporting similarities of exactly zero for both classes that we also observed for STS-B. Similarly, the values for both classes are fairly low, with only a mean similarity for the paraphrases of 0.39. However, the visual analysis shows that this mean value is somewhat skewed by the many values that are exactly zero, as the peak for the non-zero similarities is rather around 0.6. Moreover, both the distribution as well the lower mean value (0.26) indicate that the negative examples receive lower similarity scores, as expected. Still, based on the visual analysis, the distributions of the positive and negative samples strongly overlap, meaning that while the score trends in the expected direction at scale, it is not suited for individual results or calibrated as would be expected for this data.

For QQP, the results for all three models are comparable with respect to their alignment with out expectations to the MRPC: STSScore matches our expectations very well. We observe both a lot of very high values, and overall a rather high for the duplicate questions. The mean is a bit lower than for MRPC. However, this rather seems to be a property of analyzing questions in comparison to regular sentences, as we observe such a downward shift across all classes and similarity measures. We also observe the expected trend towards very low values. S-BERT is a bit a mixed bag for QQP. One the one hand, the results for the duplicate questions indicates a better measurement of the similarity than for STSScorer. On the other hand, the negative examples also receive higher similarity scores of the same amount and are also having their peak at a very high similarity of 0.8, though the distribution is spread out such that it is almost uniform between about 0.5 and about 0.8. When we consider these results in the context of the other data sets, this can be explained by the general property of S-BERT to produces higher values for the measurement of the similarity. We note that S-BERT has seen some of the data from QQP during the pre-training, which may be the reason for the very high similarity of the duplicates. This advantage not withstanding, it seems that STSScore and S-BERT are comparable on the QQP data, with a stronger separation observed with STSScore, but higher similarities for duplicates with S-BERT.

BERTScore is again exhibit a lot of the expected properties, but again fails to achieve very large values and has an overall lower similarity for the duplicate questions than STS-B. Same as above, this leads to a larger overlap between the distributions of the classes. The tendency to produce values of exactly zero is strongest for the QQP data. In general, one can say that BLEU failed for this data set: while there is still some difference in the mean value, most similarity scores of BLEU are exactly zero for both classes.

5 Discussion

Our most important result is that our experiments support our hypothesis: for all three data sets, the transformer-based prediction approach STSScorer aligned best with the expectation from the data and we suggest to use such approaches for future calculations of the semantic similarity. Especially the S-BERT approach also yields good results, though it seems to have a tendency to overestimate the similarity, even on the QQP data which was partially seen during training. BERTScore also yields promising results, but the scores are only moderately correlated with human labeled data (as measured with STS-B) and fail to fully capture semantic equality (as shown with MRPC and QQP). As could be expected, older n-gram based approaches like BLEU are not comparable and yield significantly worse results: the distribution of the similarity scores is overall fairly random, although the general trend of the labels can still be (weakly)

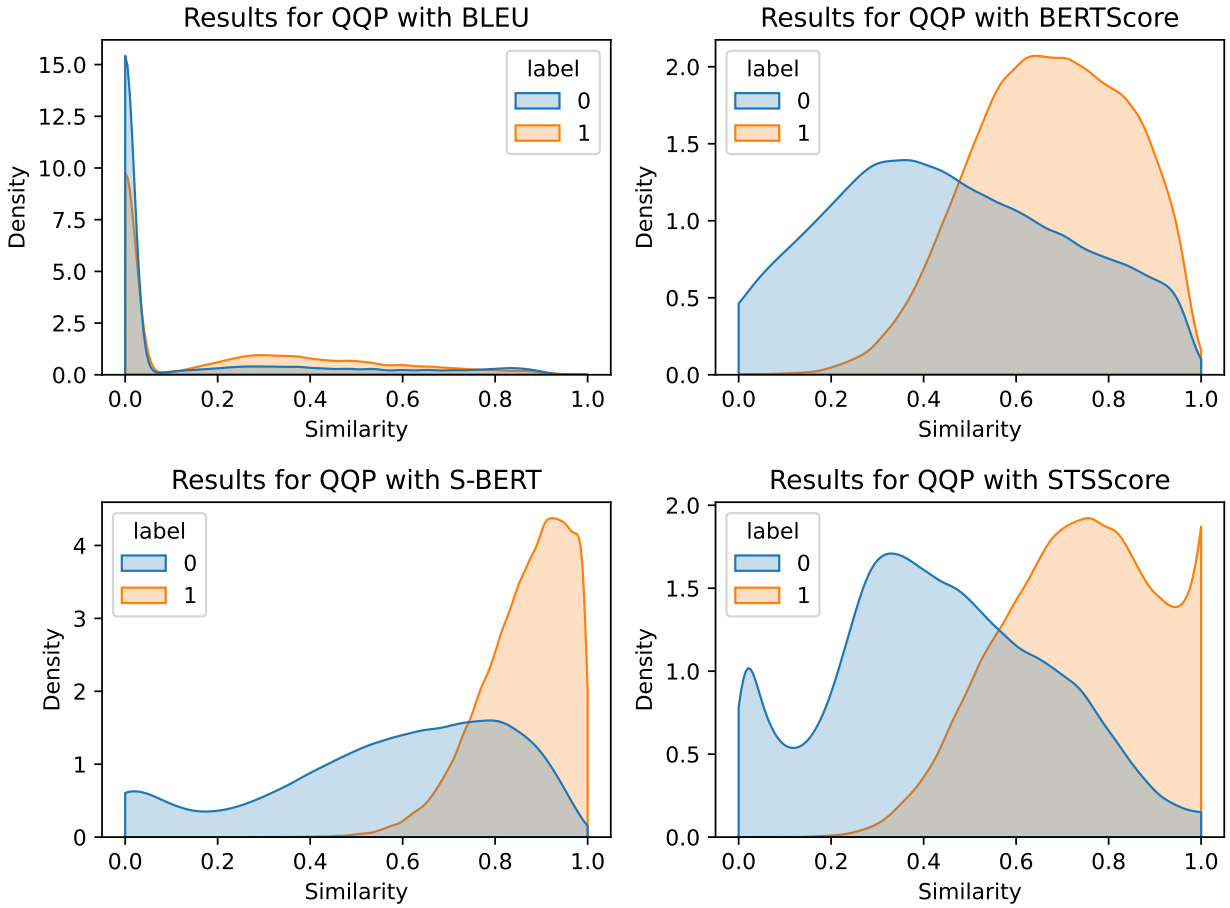

Figure 3: Evaluation of similarity measures on the training data of QQP. Ideally, the positive class (1) has scores close to one and the negative class (0) has smaller values, with only a small fraction being close to zero.

observed. This means that ranking based on BLEU are likely not misleading, if a large amount of data is used and the effects between different models are large.

Another important aspect that we want to stress is that the consideration of the actual distributions of the similarities computed by the different models gave us insights well beyond the statistical markers. Just looking at the statistics, we would have missed important aspects like the tendency of BLEU towards values of exactly zero, the tendency of BERTScore to not yield very large values close to one semantically equal statements, or the differences between STSScorer and S-BERT on paraphrases within the MRPC data.

We further note that our goal was to evaluate our hypothesis and provide an easy method for better semantic scoring. As a consequence, we choose to simply re-use an existing model from Huggingface instead of training our own model or using a checkpoint from elsewhere. While the model we use is not a lot worse then the top performers in the GLUE benchmark (best performing models in STS-B achieve correlations of up to 0.93), it cannot be considered as state-of-the-art for STS-B. Nevertheless, our results hold, i.e., STSScorer is a very good approach, though plugging in other models fine-tuned for semantic similarity prediction might yield (slightly) better results. We also note that Reimers & Gurevych (2019) also considered a fine-tuned version of S-BERT on the STS-B data by using the STS-B scores divided by five (same as us) to define a cosine similarity loss, which seemed to improve the alignment with STS-B when computing the cosine similarity. Nevertheless, based on the results reported by Reimers & Gurevych (2019), using the predictions directly should still be a better similarity measure than computing the cosine similarity of the embeddings. However, embeddings have the advantage, that they also enable other kinds of analysis, e.g., clustering or visualizations.

Even though we believe that STSScorer is currently the best method for computing similarities, there is also an important drawback: when we use transformer-based approaches to evaluate other – likely transformer-based – approaches, the evaluation will possibly have all the problems regarding biases that transformers have. Thus, we may further encode these biases, because we selected models based on a biased evaluator. Additionally, all the current work ignores that semantic similarity is not absolute and also depends on the perspective of humans and can, e.g., be influenced by culture, social background, and other aspects, raising the question whether we should rather specify our notion of similarity more carefully in the future and use multiple similarity measures that can capture such differences. Nevertheless, current technologies do not provide a better solution, though we suggest that future work should also consider ethical and fairness concerns of semantic similarity measures to understand – and hopefully mitigate – such problems.

6 Conclusion

The jumps in performance for natural language processing enable us to directly predict the semantic similarity instead of using embedding-based approaches or heuristics. Due to the readily available models on platforms like Huggingface, switching to such models for the future evaluation of the semantic similarity of results should be easily possible.

Broader Impact Statement

For many low-level tasks, we now have very good classification and regression models. We should consider using these models more often as performance measures for more complex task. We demonstrate this in this paper by showing that fine-tuned semantic similarity models are better at capturing expected characteristics of semantic similarity than relying only on the cosine-similarity of embeddings. However, when doing this, we risk that biases and limitations from the fine-tuned model affect the evaluation of complex task in an adversarial manner, which should also be further studied.

References

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 1–14, Vancouver, Canada,

August 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2001. URL https://aclanthology.org/S17-2001.

Marco Del Tredici and Núria Bel. A word-embedding-based sense index for regular polysemy representation. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pp. 70–78, Denver, Colorado, June 2015. Association for Computational Linguistics. doi: 10.3115/v1/W15-1510. URL https://aclanthology.org/W15-1510.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

William B. Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005. URL https://aclanthology.org/I05-5002.

William Held. https://huggingface.co/willheld/roberta-base-stsb, 2022.

Shankar Iyer, Nikhil Dandekar, and Kornel Csernai. First quora dataset release: Question pairs, 2017. URL https://quoradata.quora.com/First-Quora-Dataset-Release-Question-Pairs.

Alon Lavie and Abhaya Agarwal. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pp. 228–231, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://aclanthology.org/W07-0734.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

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

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Nils Reimers. https://huggingface.co/sentence-transformers/all-minilm-l6-v2, 2021.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. *arXiv preprint arXiv:1706.09799*, 2017.

Hiroki Shimanaka, Tomoyuki Kajiwara, and Mamoru Komachi. RUSE: Regressor using sentence embeddings for automatic machine translation evaluation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pp. 751–758, Belgium, Brussels, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6456. URL https://aclanthology.org/W18-6456.

Miloš Stanojević and Khalil Sima'an. BEER: BEtter evaluation as ranking. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pp. 414–419, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-3354. URL https://aclanthology.org/W14-3354.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/4496bf24afe7fab6f046bf4923da8de6-Paper.pdf.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SkeHuCVFDr.