Testing Paraphrase Models on Recognising Sentence Pairs at Different Degrees of Semantic Overlap

Qiwei Peng David Weir Julie Weeds
University of Sussex
Brighton, UK
{qiwei.peng, d.j.weir, j.e.weeds}@sussex.ac.uk

Abstract

Paraphrase detection is useful in many natural language understanding applications. Current works typically formulate this problem as a sentence pair binary classification task. However, this setup is not a good fit for many of the intended applications of paraphrase models. In particular, such applications often involve finding the closest paraphrases of the target sentence from a group of candidate sentences where they exhibit different degrees of semantic overlap with the target sentence. To apply models to this paraphrase retrieval scenario, the model must be sensitive to the degree to which two sentences are paraphrases of one another. However, many existing datasets ignore and fail to test models in this setup. In response, we propose adversarial paradigms to create evaluation datasets, which could examine the sensitivity to different degrees of semantic overlap. Empirical results show that, while paraphrase models and different sentence encoders appear successful on standard evaluations, measuring the degree of semantic overlap still remains a big challenge for them.

1 Introduction

Detecting paraphrases is useful in many natural language understanding applications, such as question answering (Yin et al., 2015; Gan and Ng, 2019), fact checking (Jiang et al., 2020), and text summarisation (Kryściński et al., 2018, 2019). Researchers have constructed paraphrase identification benchmarks, typically formulating the problem as a sentence pair classification task (Dolan and Brockett, 2005; Lan et al., 2017; Iyer et al., 2017; Zhang et al., 2019b).

Sentence pairs that have the same or largely equivalent semantics are considered as paraphrases of each other (Androutsopoulos and Malakasiotis, 2010; Bhagat and Hovy, 2013). For example:

a) More than half of the songs were purchased as albums, Apple said.

b) Apple noted that half the songs were purchased as part of albums.

Not only is it unclear what the criteria is for determining when a sentence pair has sufficiently similar semantics to be considered paraphrases, but as Chen et al. (2020) point out, the standard paraphrase classification task is not a good fit for many of the intended applications of paraphrase models. In particular, such applications are often retrieval tasks that involve finding the closest paraphrases of some target sentences within a set of documents, where candidate sentences exhibit different degrees of semantic overlap with the target sentence. To apply models to a paraphrase retrieval scenario, a paraphrase model must be sensitive to the degree to which two sentences are paraphrases of one another.

We use the term partial paraphrase to refer to situations where a sentence pair has **some** overlap in meaning, but this can range from nearly exact paraphrases to pairs that share very little meaning. An example of an intermediate case is given below:

a) More than half of the songs were purchased as albums, Apple said yesterday in a meeting with Sony.

b) Apple noted that half the songs were purchased as part of albums.

The setup used for standard paraphrase classification can be adapted to the partial paraphrase task, where the softmax confidence score is used as an estimate of the degree to which two sentences are paraphrases of one another. Indeed, this has been used in ranking tasks across different domains (MacAvaney et al., 2019; Ji et al., 2020; Sun and Duh, 2020). However, while pre-trained language models have shown good performance on the standard classification task (Devlin et al., 2019; Liu et al., 2019), as we will show, these models are often fooled by partial paraphrases where there is significant, but far from complete, semantic over-

*Proceedings of the The 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 259–269

lap.

Current paraphrase identification datasets do not test models in a partial paraphrase ranking setup. Though the semantic textual similarity (STS) tasks (Agirre et al., 2012; Cer et al., 2017) exhibit similarities to this setup as they also try to measure gradations of meaning overlap, there are some significant differences. Firstly, the ranking setup in STS concerns comparing completely different sentence pairs (e.g., $(a, b) > (c, d)$), while most paraphrase applications aim to compare different sentences with the same pivot sentence (e.g., $(a, b) > (a, c)$). Secondly, as Wang et al. (2022) point out that the definition of similarity in STS is rather vague and various complicated relations between sentence pairs all contribute to the similarity score. The difference in the similarity score cannot guarantee the different degree of semantic overlap.

Our aim, in this paper, is to rectify this deficiency. We draw inspiration from previous adversarial testing works utilising word swapping and number replacement (Zhang et al., 2019b; Wang et al., 2021) to produce negative examples. In this work, we propose adversarial paradigms (multiple word swap) to create evaluation datasets that consist of high-quality partial paraphrase pairs with graded semantic overlap. We aim to test whether the paraphrase score produced by existing paraphrase models and sentence encoders is a good reflection of the degree of semantic overlap. In contrast to their strong performance on standard paraphrase classification tasks, our analysis reveals that measuring the degree of semantic overlap still remains a challenge.

Our main contributions are as follow. First, in Section 3, we follow the standard fine-tuning strategy to produce two paraphrase models and then demonstrate their good performance on standard evaluation tasks and insensitivity to partial paraphrases. We then present (in Section 4) evaluation datasets which consist of high-quality partial paraphrase pairs with graded semantic overlap, constructed by multiple word swapping. We further show (in Section 5) that the distinction between partial paraphrase and exact paraphrase is a challenge for paraphrase models, and that their paraphrase scores are not a good reflection of the degree of semantic overlap. Finally, our work demonstrates that similarity scores produced by sentence encoders, though being widely used as a measure of similarity in meaning, are dominated by the degree of lexical overlap, and are poor estimators of the degree to which sentences are partial paraphrases.

2 Related Work

The definition of paraphrase has been long debated, as have the characteristics of paraphrase pairs (Androutsopoulos and Malakasiotis, 2010; Bhagat and Hovy, 2013; Rus et al., 2014; Liu et al., 2022). A widely accepted definition is that two sentences should exhibit the same or largely equivalent semantics, which suggests a bi-directional entailment relation. As Madnani and Dorr (2010) pointed out, paraphrases may occur at different levels, such as word-level, phrase-level, and sentence-level. Although there has been some work that concerned the identification of lexical and phrasal paraphrases (Ganitkevitch et al., 2013; Pavlick et al., 2015), most recent work on paraphrase identification has been performed at the sentence level, and has involved determining whether a given sentence pair is a paraphrase or not in a classification setup (Dolan and Brockett, 2005; Fernando and Stevenson, 2008; Xu et al., 2014; Zhang et al., 2019b; Liu et al., 2022).

However, paraphrase detection has been utilised in other NLP tasks. In question answering, Dong et al. (2017) utilised paraphrase detection to discover most probable paraphrases of a given question from a group of potential paraphrases by comparing their paraphrase scores. Similarly, Wang et al. (2020) integrated paraphrase detection in a information retrieval system to select the best paraphrased queries which are used to expand the original query list. Accordingly, Chen et al. (2020) argued that the standard binary classification setup of paraphrase identification is ill-suited to many real-world applications which involve paraphrase retrieval. To apply paraphrase models to such a retrieval scenario, the model must be sensitive to the degree to which two sentences share semantic content.

Though pre-trained language models show good performance when fine-tuned on paraphrase identification datasets (Devlin et al., 2019; Liu et al., 2019; Arase and Tsujii, 2021), a performance drop is often observed when being tested for robustness under different adversarial scenarios. Zhang et al. (2019b) utilised word swapping and back-translation to produce adversarial examples. Yang et al. (2019) adopted the same approach to produce adversarial pairs in a multilingual scenario. Shi

and Huang (2020) modified shared words to produce both positive and negative pairs. Wang et al. (2021) additionally proposed a robustness evaluation platform which can perform different transformations to sentence pairs, including word swapping, template-based generation and number replacement. Nighojkar and Licato (2021) employed paraphrase generators to produce sentence pairs that are both lexically and syntactically disparate. Such transformations can create partial paraphrases in different types. However, these partial paraphrases do not exhibit decreasing semantic overlap. To measure the sensitivity of models to different degrees of semantic overlap, we draw inspiration from them and create a list of partial paraphrases with decreasing semantic overlap for each paraphrase pair.

3 Background and Preliminaries

The classification setup for the evaluation of paraphrase identification involves identifying whether the given sentence pair is a paraphrase or not. In this section, we follow previous work and first fine-tune BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) on widely used paraphrase datasets to produce standard paraphrase models and check whether their success on standard evaluation benchmarks could transfer to the recognition of partial paraphrases with different degrees of semantic overlap.

3.1 Datasets

In this paper, we mainly consider two commonly used paraphrase datasets, $\text{PAWS}_{\text{Wiki}}$ and PAWS_{QQP} (Zhang et al., 2019b). The Paraphrase Adversaries from Word Scrambling (PAWS) datasets utilise word scrambling (swapping words that have same part-of-speech or name entity tags) and back translation to produce both positive and negative examples for given sentences while maintaining high lexical overlap. Though less often used than datasets like Microsoft Research Paraphrase Corpus (MRPC) (Dolan and Brockett, 2005) where a large percentage of positive sentence pairs just have partial overlap in meaning, PAWS datasets contain high quality sentence pairs that are mostly exact paraphrases. PAWS datasets do not have a specific license and can be used freely for any purpose[1]. In the following sections, we propose adversarial evaluation datasets that are derived from the test sets of these two datasets.

Datasets	Train	Dev	Test
PAWS_{QQP}	11,986	-	677
$\text{PAWS}_{\text{Wiki}}$	49,401	8,000	8,000

Table 1: Statistics of two PAWS datasets.

The statistics of these datasets are listed in Table 1. Below we give some brief descriptions:

- **PAWS_{QQP}**: With the aim of assessing sensitivity to word order and syntactic structure, Zhang et al. (2019b) proposed a paraphrase identification dataset that contains sentence pairs of high lexical overlap. They are created by applying back translation and word scrambling to sentences taken from the Quora Question Pairs (Wang et al., 2017).
- **$\text{PAWS}_{\text{Wiki}}$**: The same process is applied to sentences obtained from Wikipedia articles to construct paraphrase and non-paraphrase pairs.

The construction process ensures positive sentence pairs in PAWS datasets are mostly exact paraphrases.

Model	$\text{PAWS}_{\text{Wiki}}$		PAWS_{QQP}	
	ACC	F1	ACC	F1
BERT	92.31	91.59	89.07	81.95
RoBERTa	94.10	93.44	92.91	87.76

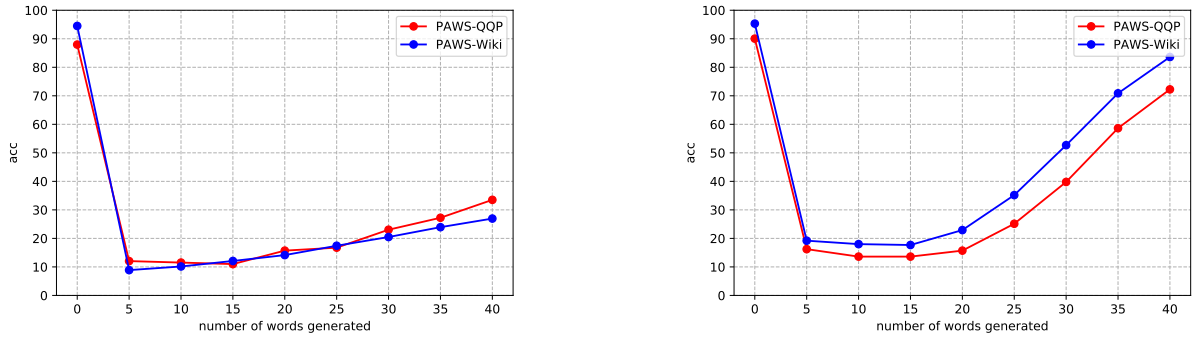
Table 2: Classification results on PAWS datasets; we report the F1 score of the positive class and the overall accuracy.

3.2 Models

We evaluate two pre-trained language models, BERT and RoBERTa[2]. They are widely used, and have achieved good performance on paraphrase identification tasks. Following previous work, we first fine-tune them on the paraphrase datasets to produce standard paraphrase models. As shown in Zhang et al. (2019b), the best performance is achieved by training on the combination of the original QQP dataset (which has 384,348 training sentence pairs), PAWS_{QQP} and $\text{PAWS}_{\text{Wiki}}$. We follow the same strategy, fine-tuning both BERT-base

[1] https://github.com/google-research-datasets/paws/blob/master/LICENSE

[2] We use their huggingface implementations: https://huggingface.co/bert-base-uncased (110 million parameters) and https://huggingface.co/roberta-base (123 million parameters)



(a) BERT

(b) RoBERTa

Figure 1: Performance of BERT and RoBERTa on generation-based adversarial evaluation datasets in the classification setup. X-axis: The number of generated words added to sentence A. Y-axis: The accuracy.

and RoBERTa-base on this combined training set[3]. Also following Zhang et al. (2019b), we use the QQP development set as our development set for early stopping. Each model is fine-tuned for 3 epochs with batch size of 16. We use the Adam optimiser with learning rate of 2e-5 and a linear learning rate warm-up over 10% of the training data. We fine-tune each model five times and choose the best for later experiments according to their performance on the development set. All of our experiments are conducted on one RTX 3090 GPU and each epoch takes around one hour.

We include the results on the standard evaluation benchmarks in Table 2. We can see that both BERT and RoBERTa have achieved high accuracy and F1 scores, which appears to demonstrate their ability to identify paraphrases.

3.3 Partial Paraphrases

A typical example of partial paraphrase is where one sentence contains all of the semantics of another but also contains additional information (see the example in the Introduction). We therefore adopt a straightforward approach to produce an initial adversarial test of partial paraphrase identification.

Given a positive sentence pair (a, b) in PAWS test sets, we take a as context and utilise the GPT2[4] generation model (Radford et al., 2019) to generate additional tokens, giving a new sentence that we denote $\hat{a}$. To avoid disrupting the meaning of the existing content, we further add ", and" to the end of a. Compared to the original pair (a, b), the new pair $(\hat{a}, b)$ has lower semantic overlap given the additional information in $\hat{a}$.

Here, we give an example of generated partial paraphrase pairs:

a) He was born in New York City in East Broadway on October 23, 1806**, and was raised in Baltimore]], Maryland, where the family moved]] to live in 1900 with]] two sons and two daughters.]]**

b) He was born on 23 October 1806 in New York, East Broadway.

The bold part is the generated text, and the coloured "]]" symbols indicate places where we truncate the added content (every five generated tokens). The idea is that the dataset contains a range of examples that systematically vary in terms of the degree of semantic overlap. We evaluate previously fine-tuned paraphrase models on this generation-based evaluation set (no further training) and investigate at what point they detect that the given pair is no longer an exact paraphrase.

Experimental results are summarised in Figure 1, where we report the overall accuracy. We observe that when no extra words are added, these two models show near-perfect performance on recognising the given positive pair as paraphrase of each other. However, when we add 5 words to produce a partial paraphrase pair as a negative example of a paraphrase, performance drops dramatically, demonstrating the lack of sensitivity of these models to the distinction between an exact paraphrase and a partial paraphrases. The accuracy gradually increases as we generate more words to sentence

[3]We also tried training on individual datasets rather than the combined one. The results show worse performance on the standard classification evaluation and no different trend on following ranking tasks.

[4]We choose GPT2 because it generates satisfactory results and is free to access.

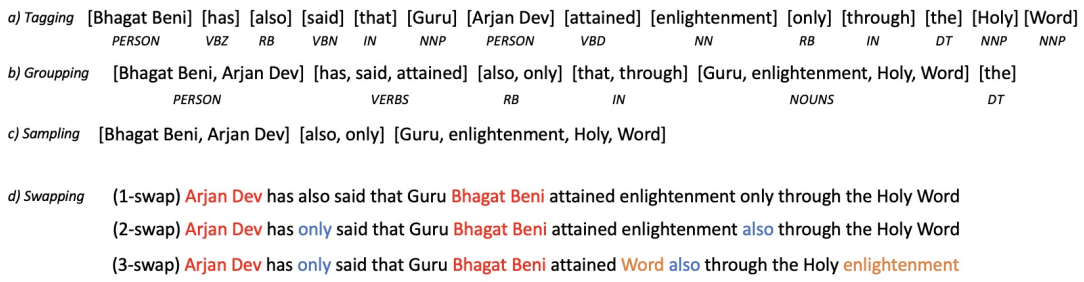


Figure 2: Illustration of the multi-swap method in four steps. a) Tag words and phrases with part-of-speech (POS) and named entities. b) produce candidate sets by grouping words and phrases with the same tag. c) Sample three groups from the candidate sets that have two or more words/phrases. d) Swap position.

A. However, the increase only becomes substantial when we append at least than 20 words. We can see that good performance on the original test sets is not translating to the task of distinguishing partial paraphrases from exact paraphrases.

Though paraphrase models are can be fooled by partial paraphrases, they do exhibit increased ability to recognise them as the difference in semantics grows. The poor performance on close partial paraphrases might be explained by the paraphrase score decreasing as the degree of semantic overlap reduces, but the decrease not being large enough to bring the score down below the binary classification threshold, resulting in the wrong prediction. To explore whether this is the case and whether the paraphrase score could act as a reliable indicator to the degree of semantic overlap, we now turn to the evaluation of paraphrase models in a ranking scenario, requiring candidates to be ranked based on the amount of semantic overlap.

Problematically, however, sentences produced by the generation-based method exhibit significant differences in sentence length as well as the degree of lexical overlap. These differences would be an obvious clues in a ranking task[5]. In this regard, we adopted a different approach to produce ranking-based evaluation datasets which was to utilise word swapping.

4 Partial Paraphrase Construction

To create partial paraphrases at decreasing degrees of semantic overlap, while maintaining lexical overlap and sentence length, we draw inspiration from Zhang et al. (2019b) and Wang et al. (2021) who create negative examples by swapping words and entities. We take positive sentence pairs from PAWS test sets and create corresponding partial paraphrases with graded semantic overlap by making multiple word swaps. Since the semantics are equivalent for positive sentence pairs, we always make modifications to sentence B to produce partial paraphrase variants and compare them with the original sentence A. This can increase the task difficulty as the lexical overlap will be high for negative pairs. Models that produce high scores based on high lexical overlap are likely to fail in this scenario.

	# original	# after 3 swaps
$\text{PAWS}_{\text{Wiki}}$	3536	1382
PAWS_{QQP}	191	63

Table 3: The number of examples before and after performing 3 swaps. We take only positive examples (3536/191) from original datasets and filter out sentence pairs that do not meet our criteria as described in the construction process. We end up with 1382/63 positive examples and each now has 3 swap-based negative variants.

Figure 2 illustrates the multi-swap procedure. Given a paraphrase pair (a, b), we first perform part-of-speech tagging[6] (POS) on b to obtain tags for each word. We further detect named entities like locations, person names, organisations, and dates using a named entity tagger, and replace POS tags with entity tags when there is overlap. Words and phrases that have the same tag[7] are then grouped together. We deduplicate each group to avoid swapping the position between two identical words/phrases. Given that a swap requires at least

[5]Our initial experiments show that sentence encoders can achieve extremely high performance on ranking these sentence pairs by capturing such clues.

[6]We use Spacy large web-based model pipeline (en_core_web_lg) for both POS and NER tagging.

[7]We do not distinguish different POS tags for verbs (e.g., VBZ, VBN, VBD) and nouns (e.g., NNP, NNPS, NN, NNS). We also exclude "to be" verbs, as swapping them does not guarantee changes in semantics.

Source	Sentence A	Sentence B	Paraphrase Degree
$\text{PAWS}_{\text{Wiki}}$	**(no-swap)** Bhagat Beni also said that the guru Arjan Dev has obtained enlightenment only through the Holy Word.	Bhagat Beni has also said that Guru Arjan Dev attained enlightenment only through the Holy Word.	4
	(1-swap) Arjan Dev has also said that Guru Bhagat Beni attained enlightenment only through the Holy Word.		3
	(2-swap) Arjan Dev has only said that Guru Bhagat Beni attained enlightenment also through the Holy Word.		2
	(3-swap) Arjan Dev has only said that Guru Bhagat Beni attained Word also through the Holy enlightenment.		1
PAWS_{QQP}	**(no-swap)** Was increasing funding to protect Benghazi before the attack denied by Congress. If so, who voted against it?	Was increased funding to protect Benghazi before the attack denied by Congress. If so, who voted against it?	4
	(1-swap) Was increased attack to protect Benghazi before the funding denied by Congress. If so, who voted against it?		3
	(2-swap) Was increased attack to protect Congress before the funding denied by Benghazi. If so, who voted against it?		2
	(3-swap) Was denied attack to protect Congress before the funding increased by Benghazi. If so, who voted against it?		1

Table 4: Examples of swapped sentences taken from two PAWS datasets (We swap sentence B to produce swap-based partial paraphrases). Different colours denote different swaps and each swap is performed based on previous swaps to ensure the degrading semantic overlap. Sentence pair with paraphrase degree of 4 is **exact paraphrase** and 3, 2, 1 are **partial paraphrases** with decreasing semantic overlap.

two words/phrases, we discard tag groups that have less than two words/phrases. In order to produce enough candidates for ranking, we filter out sentences that have less than 3 tag groups. For each sentence with at least three tag groups, we randomly sample three groups, and from each group we randomly sample two words/phrases. In the end, we swap the position of sampled words/phrases to produce swapped sentences. We perform each swap based on previous swaps, with a maximum of three swaps. In summary, given a positive sentence pair (a, b), we apply our multi-swap strategy on b, and produce a group of sentence pairs $[(a, b), (\hat{b}_{1swap}, b), (\hat{b}_{2swap}, b), (\hat{b}_{3swap}, b)]$, where they exhibit decreasing semantic overlap.

The statistics of the resulting evaluation datasets are given in Table 3. Examples taken from the swap-based partial paraphrase datasets are shown in Table 4. In the same group, sentences with higher paraphrase degree are more likely to be exact paraphrases. Our evaluation setup is as follow: given a paraphrase scoring function f, and a set of sentence pairs $\{(a, b), (\hat{b}_{1swap}, b), (\hat{b}_{2swap}, b), (\hat{b}_{3swap}, b)\}$. We expect $f(a, b) > f(\hat{b}_{1swap}, b) > f(\hat{b}_{2swap}, b) > f(\hat{b}_{3swap}, b)$.

5 Experiments

We compare previous fine-tuned paraphrase models (BERT and RoBERTa in Section 3) with sentence encoders. Sentence encoders, such as SBERT(Reimers and Gurevych, 2019) and SimCSE (Gao et al., 2021), are widely used in various ranking scenarios which aim to measure the similarity in meaning between two sentences. They use a contrastive learning objective, intended to derive high-quality sentence representations by pulling sentences with similar semantics closer together and pushing dissimilar ones apart. Although they have achieved relatively good performance on STS tasks, it is unclear whether the similarity score they produce can be used to measure the extent to which sentence pairs are paraphrases.

In this experiment, we evaluate SimCSE[8], two variants of SimCSE, namely, SimCSE+PAS (Peng et al., 2022) and SimCSE+BERTScore[9] (Zhang et al., 2019a), and two SBERT models[10] (Reimers and Gurevych, 2020) which are specifically trained on paraphrase datasets. We denote

[8]https://github.com/princeton-nlp/SimCSE
[9]https://github.com/Tiiiger/bert_score
[10]https://github.com/UKPLab/sentence-transformers

one as SBERT_{v1}[11] and the other as SBERT_{v2}[12].

For paraphrase models, we use its softmax confidence of being positive as the **paraphrase score** to rank sentence pairs. For sentence encoders, we rank sentence pairs using their default strategy to produce a paraphrase score. Specifically, SBERT and SimCSE utilise the cosine similarity between two sentences; SimCSE+PAS increases the interaction between two sentences by considering the aggregated score over predicate-argument alignments; and SimCSE+BERTScore considers the IDF-weighted F1 measure in terms of word matching.

5.1 Evaluation

The ranking results are summarised in Table 5. We report both the average R-Precision and the average Spearman rank correlation between the predicted ranking and the true ranking across all groups. R-Precision measures the ability to retrieve best paraphrases and Spearman rank correlation measures the overall sensitivity to different degrees of semantic overlap as it concerns relative position shifts in the group. Similarly, we can turn this ranking task into a classification problem by regarding sentence pairs with paraphrase degree of 4 as positive and sentence pairs that have lower degree as negative. In this setup, we only evaluate paraphrase models. The classification results are shown in Figure 3.

From Figure 3, we observe similar patterns as in previous generation-based classification experiments. Both BERT and RoBERTa show good performance on recognising the given pair as paraphrases when no-swap is applied. However, after we perform one swap, the performance drops significantly, showing that these models fail to recognise the distinction. Both models begin to recover from this situation after two swaps[13]. This, again, indicates that paraphrase models are often confused by small semantic differences in the classification setup.

In terms of the ranking results presented in Table 5, we can see that sentence encoders show limited ability to distinguish the exact paraphrase from partial paraphrases on $\text{PAWS}_{\text{Wiki}}$, which is evidenced by the low R-Precision score. Although their overall performance is higher on PAWS_{QQP}, we suspect this is due to the high lexical overlap, which we investigate in detail in Section 5.2. Compared to sentence encoders, paraphrase models show generally better performance in terms of R-Precision on both datasets. It is worth noting that, under the classification setup, paraphrase models achieve good accuracy on recognising non-swap positive pairs (see the high accuracy of 0-swap in Figure 3). However, when we mix the non-swap pair together with other swapped partial paraphrases, both BERT and RoBERTa are unable to achieve equivalent R-Precision scores. This shows that paraphrase scores produced for partial paraphrases are often higher than those for exact paraphrases, demonstrating that they are not a reliable indicator as to how close two sentences are to being paraphrases. Since the number of candidates to rank (only four sentence pairs in each group) is small, the Spearman rank correlation obtained by both models is insufficient to demonstrate a strong positive correlation and implies many position shifts in the predicted ranking. Although the two versions of the SBERT model are specifically trained on paraphrase datasets, they do not exhibit better performance than SimCSE.

Model	$\text{PAWS}_{\text{Wiki}}$ (swap)		PAWS_{QQP} (swap)	
	RPrec	Spearman	RPrec	Spearman
BERT	77.57	73.42	69.84	78.10
SimCSE	41.97	57.68	71.43	83.81
SimCSE+PAS	48.99	69.71	63.93	81.64
SimCSE+BERTScore	42.33	71.43	66.67	88.89
RoBERTa	85.31	69.90	76.19	69.21
SimCSE	43.20	56.98	66.67	79.05
SimCSE+PAS	42.19	62.13	62.30	83.61
SimCSE+BERTScore	44.21	69.90	71.43	87.62
SBERT_{v1}	35.60	47.97	74.60	81.90
SBERT_{v2}	28.44	37.77	66.67	77.46

Table 5: The results on the swap-based ranking evaluation. The backbone of sentence encoders in the first block is BERT-base and RoBERTa-base in the second block. We report the R-Precision and Spearman correlation.

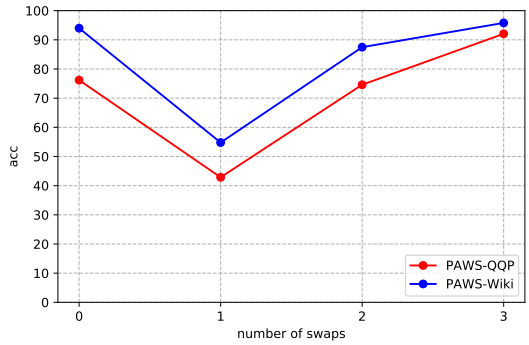
5.2 The Impact of Lexical Overlap

One observation we have from Table 5 is that all sentence encoders have higher R-Precision and Spearman correlation on PAWS_{QQP} compared to the performance on $\text{PAWS}_{\text{Wiki}}$. As shown in Table 6, we can see that positive sentence pairs in PAWS_{QQP} have significantly higher lexical overlap. Thus, we suspect that the higher lexical overlap

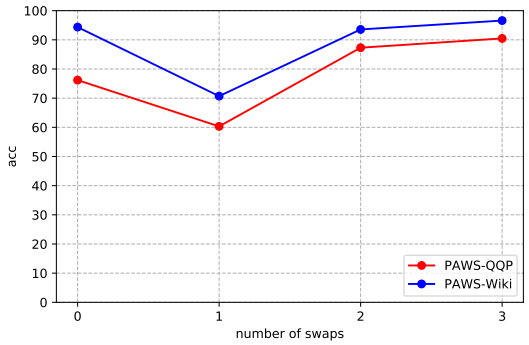
[11] sentence-transformers/paraphrase-MiniLM-L12-v2

[12] sentence-transformers/paraphrase-distilroberta-base-v2

[13] As we increase the number of swaps, models become more confident in distinguishing whether the sentence pair is a paraphrase or not. This trend also reflects the quality of swap-based examples we create.



(a) BERT



(b) RoBERTa

Figure 3: Performance of BERT and RoBERTa on swap-based evaluation datasets in the classification setup. X-axis: Number of swaps performed. Y-axis: The accuracy. For Swap-0, all sentence pairs are positive and the accuracy is the percentage of sentence pairs classified as paraphrases. For Swap 1 to 3, sentence pairs are now all turned into negatives and the accuracy is the percentage of sentence pairs correctly classified as non-paraphrases by the model after we perform different word swaps.

	Lexical Overlap
$\text{PAWS}_{\text{Wiki}}$ (swap)*	83.46%
- after back-translation	75.79%
PAWS_{QQP} (swap)	95.03%
- after back-translation	59.56%

Table 6: The lexical overlap of the positive sentence pair (pair of paraphrase degree of 4). * denotes the randomly sampled dataset. We calculate the lexical overlap in terms of Jaccard Similarity with ngram=1.

Model	$\text{PAWS}_{\text{Wiki}}$ (swap) (100sample-bt)		PAWS_{QQP} (swap) (bt)	
	RPrec	Spearman	RPrec	Spearman
BERT	67.00	71.00	65.08	75.56
SimCSE	31.00	48.40	31.75	54.60
SimCSE+PAS	41.00	61.20	33.33	59.68
SimCSE+BERTScore	31.00	61.60	30.16	58.73
RoBERTa	74.00	69.20	73.02	68.57
SimCSE	33.00	47.80	33.33	55.56
SimCSE+PAS	32.00	51.20	36.51	57.14
SimCSE+BERTScore	30.00	56.00	26.98	53.02
SBERT_{v1}	26.00	38.20	28.57	53.97
SBERT_{v2}	15.00	17.20	4.76	7.30

Table 7: The results on the swap-based ranking evaluation (back-translated). We report the R-Precision and Spearman correlation.

makes sentence encoders produce higher scores which enable them to "guess" the correct answer.

To verify the impact of lexical overlap, we apply back-translation[14] to the positive sentence A so that the positive pair now has much lower lexical overlap. Given the PAWS_{QQP} (swapped) is of small size (63 groups), we manually check the results of back-translation and correct them if the translated sentence A is no longer an exact paraphrase of sentence B. $\text{PAWS}_{\text{Wiki}}$ (swapped) has more then 1,300 groups of sentence pairs, so we randomly sample 100 groups from it and apply the same process. As shown in Table 6, the lexical overlap has been significantly reduced after we apply back translation.

We evaluate all models on the back-translated datasets and the results are presented in Table 7. After reducing lexical overlap for positive pairs, we observe performance drops for all models. In particular, both R-Precision and Spearman rank correlation have decreased significantly across all sentence encoders. This indicates that sentence encoders are largely affected by lexical overlap while BERT and RoBERTa seem more robust to different degrees of lexical overlap between two sentences. Furthermore, we see that the performance of both predicate-argument alignment (PAS) and word matching (BERTScore) is only slightly better than that of SimCSE in terms of the sensitivity to semantic overlap. This demonstrates that the changes in similarity scores they produce are not good measurements as to how close two sentences are to being paraphrases. Given the unsatisfactory performance of paraphrase models and sentence encoders, we stress that more efforts are necessary to improve models' sensitivity to different degrees of semantic overlap, and it is important to consider specific ranking objectives and the proximity be-

[14]We utilise the Marian machine translation model (Junczys-Dowmunt et al., 2018) and use German as the pivot language.

tween different sentence pairs.

6 Conclusion

In this paper, we explore whether paraphrase scores produced by paraphrase models and sentence encoders are reliable indicators of the degree to which two sentences share semantic content. Accordingly, we propose an adversarial paradigms (multiple word swap) to create evaluation datasets that consist of high-quality partial paraphrases with graded semantic overlap in a ranking setup. Our experimental results show that the similarity score produced by sentence encoders is not a good indicator of how close two sentences are to being exact paraphrases, and is heavily affected by lexical overlap. Whilst paraphrase models show generally better performance, the confidence scores they produce are still far from acting as a reliable indicator to different degrees of semantic overlap. Measuring the degree of semantic overlap between two sentences remains a significant challenge. Our future work includes producing larger ranking datasets and extending this paradigm to other relevant datasets.

Limitations

The remaining limitations in our work are two-fold. First, for specific paraphrase models, our experiments are limited to consideration of BERT-base and RoBERTa-base models. This choice is made following their generality and good performance on various NLP tasks, but larger language models could also be considered. The second limitation of this paper is that, under the swap-based strategy, the sentence after three swaps sometimes are semantically problematic though grammatically correct. Despite having shown that paraphrase models have improved ability to distinguish partial paraphrases after two swaps, it would be better to use naturally occurring sentences and reduce the clue of irregular word or phrase usages.

Acknowledgement

We thank all anonymous reviewers for their insightful comments. We would like to further thank Bowen Wang for helpful discussions and proofreading.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In ** SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393.

Ion Androutsopoulos and Prodromos Malakasiotis. 2010. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, 38:135–187.

Yuki Arase and Junichi Tsujii. 2021. Transfer fine-tuning of bert with phrasal paraphrases. *Computer Speech & Language*, 66:101164.

Rahul Bhagat and Eduard Hovy. 2013. What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

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

Hannah Chen, Yangfeng Ji, and David K Evans. 2020. Pointwise paraphrase appraisal is potentially problematic. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 150–155.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Li Dong, Jonathan Mallinson, Siva Reddy, and Mirella Lapata. 2017. Learning to paraphrase for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 875–886.

Samuel Fernando and Mark Stevenson. 2008. A semantic similarity approach to paraphrase detection. In *Proceedings of the 11th annual research colloquium of the UK special interest group for computational linguistics*, pages 45–52. Citeseer.

Wee Chung Gan and Hwee Tou Ng. 2019. Improving the robustness of question answering systems to question paraphrasing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6065–6075.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910.

Shankar Iyer, Nikhil Dandekar, Kornél Csernai, et al. 2017. First quora dataset release: Question pairs. *data. quora. com*.

Zongcheng Ji, Qiang Wei, and Hua Xu. 2020. Bert-based ranking for biomedical entity normalization. *AMIA Summits on Translational Science Proceedings*, 2020:269.

Shan Jiang, Simon Baumgartner, Abe Ittycheriah, and Cong Yu. 2020. Factoring fact-checks: Structured information extraction from fact-checking articles. In *Proceedings of The Web Conference 2020*, pages 1592–1603.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

Wojciech Kryściński, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. 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)*, pages 540–551.

Wojciech Kryściński, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1808–1817.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1224–1234.

Timothy Liu et al. 2022. Towards better characterization of paraphrases. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8592–8601.

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*.

Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. 2019. Cedr: Contextualized embeddings for document ranking. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 1101–1104.

Nitin Madnani and Bonnie J Dorr. 2010. Generating phrasal and sentential paraphrases: A survey of data-driven methods. *Computational Linguistics*, 36(3):341–387.

Animesh Nighojkar and John Licato. 2021. Improving paraphrase detection with the adversarial paraphrasing task. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7106–7116.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. Ppdb 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–430.

Qiwei Peng, David Weir, and Julie Weeds. 2022. Towards structure-aware paraphrase identification with phrase alignment using sentence encoders. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4113–4123.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners.

Nils Reimers and Iryna Gurevych. 2019. 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)*, pages 3982–3992.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Vasile Rus, Rajendra Banjade, and Mihai Lintean. 2014. On paraphrase identification corpora. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2422–2429.

Zhouxing Shi and Minlie Huang. 2020. Robustness to modification with shared words in paraphrase identification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 164–171.

Shuo Sun and Kevin Duh. 2020. Clirmatrix: A massively large collection of bilingual and multilingual datasets for cross-lingual information retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4160–4170.

Bin Wang, C-c Kuo, and Haizhou Li. 2022. Just rank: Rethinking evaluation with word and sentence similarities. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6060–6077.

Xiao Wang, Qin Liu, Tao Gui, Qi Zhang, Yicheng Zou, Xin Zhou, Jiacheng Ye, Yongxin Zhang, Rui Zheng, Zexiong Pang, et al. 2021. Textflint: Unified multilingual robustness evaluation toolkit for natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 347–355.

Xiao Wang, Craig Macdonald, and Iadh Ounis. 2020. Deep reinforced query reformulation for information retrieval. *arXiv preprint arXiv:2007.07987*.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4144–4150.

Wei Xu, Alan Ritter, Chris Callison-Burch, William B Dolan, and Yangfeng Ji. 2014. Extracting lexically divergent paraphrases from twitter. *Transactions of the Association for Computational Linguistics*, 2:435–448.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. Paws-x: A cross-lingual adversarial dataset for paraphrase identification. 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)*, pages 3687–3692.

Pengcheng Yin, Nan Duan, Ben Kao, Junwei Bao, and Ming Zhou. 2015. Answering questions with complex semantic constraints on open knowledge bases. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1301–1310.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019a. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019b. Paws: Paraphrase adversaries from word scrambling. 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)*, pages 1298–1308.