

# Striking a Balance: Alleviating Inconsistency in Pre-trained Models for Symmetric Classification Tasks

Ashutosh Kumar

Indian Institute of Science, India

ashutosh@iisc.ac.in

Aditya Joshi

SEEK, Australia

aditya.m.joshi@gmail.com,

ajoshi@seek.com.au

## Abstract

While fine-tuning pre-trained models for downstream classification is the conventional paradigm in NLP, often task-specific nuances may not get captured in the resultant models. Specifically, for tasks that take two inputs and require the output to be invariant of the order of the inputs, inconsistency is often observed in the predicted labels or confidence scores. We highlight this model shortcoming and apply a consistency loss function to alleviate inconsistency in symmetric classification. Our results show an improved consistency in predictions for three paraphrase detection datasets without a significant drop in the accuracy scores. We examine the classification performance of six datasets (both symmetric and non-symmetric) to showcase the strengths and limitations of our approach.

## 1 Introduction

Symmetric classification tasks involve two inputs and require that the model output should be independent of the order in which the two input texts are given. In other words, *the output of the classifier should be the same and the confidence score must not be significantly different*, if the inputs  $X$  and  $Y$  are instead supplied as  $Y$  and  $X$ . Paraphrase detection, multi-lingual semantic similarity are examples of symmetric classification tasks. Although attention-based (Bahdanau et al., 2015; Vaswani et al., 2017) pre-trained language models have led to significant performance gains in multiple text classification tasks, they demonstrate a peculiar erratic behaviour on symmetric classification: inconsistency. An example<sup>1</sup> of *inconsistency* for paraphrase detection is shown in Figure 1. Additional examples can be found in the Appendix (Table 4). To alleviate such an inconsistency for symmetric classification tasks, we propose a simple additional

<sup>1</sup>Note that, while this particular example is based on our fine-tuned model, it will change depending on the trained model. The overall argument is valid, nonetheless.











<b>X</b>	A provisional government or a revolutionary government has been declared several times by insurgent groups in the Philippines .	
<b>Y</b>	A revolutionary government or a provisional government has been declared several times in the Philippines by insurgent groups .	
<b>Model</b>		
<b>Input Sequence</b>		
<b>X Y</b>	 (98.6)	 (88.3)
<b>Y X</b>	 (92.2)	 (87.9)

Figure 1: Impact of reordering an example input pair ( $X$  and  $Y$ ) on standard fine-tuned BERT  and BERT-with-consistency-loss . The pair are true paraphrases.  and  denote that the model predicted them to be paraphrases and not-paraphrases, respectively. Confidence scores are reported in brackets. Details in Section 1.

*drop-in* fine-tuning objective, based on either the Kullback-Leibler (KL) or Jensen-Shannon (JS) divergence (or any  $f$ -divergence (Rubenstein et al., 2019)), to the cross-entropy loss for symmetric tasks. We refer to this as the *consistency loss*.

The main contributions of this paper are:

- Highlight inconsistency issues in symmetric classification tasks,
- Describe a consistency loss function to alleviate inconsistency, and
- Demonstrate the applicability and limitations of the loss function via qualitative and quantitative analyses on tasks from the GLUE benchmark.

Additionally, to drive future research, we have made the data and code public<sup>2</sup>.

**Note:** The problem of inconsistency can be attributed in part to the positional embedding. However, it has been shown that eliminating positional

<sup>2</sup><https://github.com/ashutoshml/alleviating-inconsistency>

embedding results in a poor performance of the model (Wang and Chen, 2020; Wang et al., 2021).

## 2 Related Work

**Pre-trained Classification Models** like BERT (Devlin et al., 2019), and RoBERTa (Liu et al., 2020) are typically fine-tuned for classification tasks using a low capacity neural network classifier connected to the pre-trained model on its first token (typically [CLS] token). We demonstrate the inconsistency in the case of symmetric classification tasks for pairs of inputs, depending on the order of inputs. To the best of our knowledge, this is the first work that incorporates task-specific nuances to ensure consistency in symmetric classification.

**Consistency Loss** has been used in style transfer tasks to minimize the distance between round-trip generation of candidates for image-to-image translation (Zhu et al., 2017) or text style transfer (Huang et al., 2020). In a similar vein, we apply consistency loss (formulated as either the Kullback-Leibler or the Jensen-Shannon divergence loss) to alleviate the inconsistency problem in symmetric tasks.

**Embedding-based Semantic Similarity Scores** based on BERT-based models like SBERT (Reimers and Gurevych, 2019; Thakur et al., 2021) can map surface form realizations to embeddings. Their performance is worse than directly using BERT-style cross-encoder models for tasks such as semantic similarity (Thakur et al., 2021). However, the primary aim of such embedding-based scorers is orthogonal and, at best, complementary to the goal of our work since we want to ensure high-performing, consistent classifiers. Similarly, an alternative for symmetric classification is to separately obtain predictions for  $(X, Y)$  and  $(Y, X)$ , and then average the confidence scores during test time. But, this is a weakly grounded, heuristic-driven approach. In general, *averaging does not rectify the mistakes made by the model, only masks it.*

## 3 Method

### 3.1 Problem Description

**A.** Given a pair of input sentences  $(X, Y)$ , label  $l_{(X,Y)}$ , and a pre-trained BERT-based model  $\mathcal{M}_{PRE}$ , the goal is to output a *reliable model*  $\mathcal{M}_{REL}$  to predict an output label for a new input pair  $(X_{test}, Y_{test})$  such that the *inconsistency* between its different ordering is minimized. While we only

Category	Datasets	Train	Val.	Test
Pairwise Symmetric	QQP	327462	40430	36384
	PAWS	49401	8000	8000
	MRPC	3302	408	366
Single Sentence	SST2	60615	6872	6734
Pairwise Non-symmetric	QNLI	99506	5463	5237
	RTE	2241	277	249

Table 1: Datasets Statistics. Please refer to Section 4.

experiment with semantic similarity (or paraphrasing), the description holds true for other symmetric relations too (such as predicting if two sentences have the same polarity).

**B.** Given a model fine-tuned on the task above  $\mathcal{M}_{REL}$ , can it help in providing a better initialization for transfer learning an empirically superior model  $\mathcal{M}'$  on other downstream tasks?

### 3.2 Setup

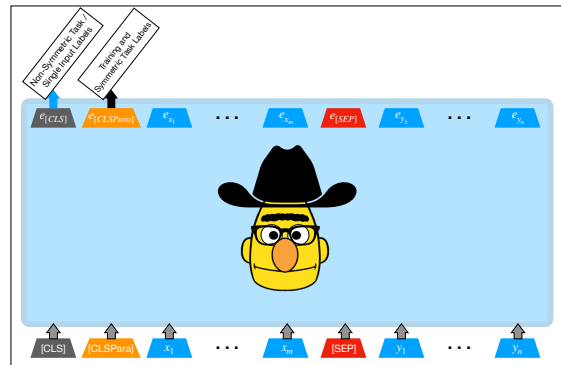


Figure 2: BERT-with-consistency-loss. We use an additional classification token: [CLSPara] for our input, upon which the consistency objective is applied. Please refer Section 3.2 for details.

For problem A (Section 3.1), the input is a concatenation of tokenized strings  $X = x_1, \dots, x_m$  and  $Y = y_1, \dots, y_n$  separated using a special token ([SEP] in the case of BERT). The concatenated inputs with the special token are passed through multiple self-attention layers (Vaswani et al., 2017). In the traditional approach, the representation of the first token ( $\langle s \rangle$  or [CLS]) is passed through a fully connected classifier layer (the same final representation is used irrespective of the arity of the task inputs). In our approach, we use the [CLSPara] representation for symmetric classification tasks whereas we use the standard first token ( $\langle s \rangle$  or [CLS]) representation for single input and non-symmetric classification tasks (Section 4). Since we first fine-tune the model on [CLSPara] representation, our approach allows for pair-wise

knowledge to be transferred to other downstream classification tasks (problem **B** (Section 3.1)).

We call this method BERT-with-consistency-loss and is shown in Figure 2. Contrasting this with a traditional BERT-based approach, we see that, in the traditional BERT-based approach, the input is pre-pended with another special symbol ([CLS] in case of BERT and <s> in case of RoBERTa). In BERT-with-consistency-loss, we concatenate an extra symbol with the special symbol. We call the extra symbol [CLSPara]. This extra token is specifically used for symmetric classification tasks to ensure consistency of prediction. The standard objective used for fine-tuning BERT-based models is the cross-entropy loss, which maximizes the probability of predicting the correct output class for a given input, given as:

$$\mathcal{L}_{ce}(y, \hat{y}) = - \sum_i y_i \log \hat{y}_i, \quad (1)$$

where  $y$  is the one-hot representation of the target class,  $\hat{y}$  is the softmax output of the model, and  $i$  is the associated co-ordinate. As described earlier, this objective may produce an inconsistent prediction based on the order of the two inputs. To overcome this weakness, we propose an additional consistency loss formulated in terms of either the KL or the JS Divergence. We pass the inputs  $X$  and  $Y$  through the same model twice, once as a pair  $(X, Y)$  (called  $L2R$ ) and then as the pair  $(Y, X)$  (called  $R2L$ ). Having obtained the outputs from the model for  $L2R$  and  $R2L$ , the final objective function for 🍌 is as follows:

$$\mathcal{L} = \mathcal{L}_{ce}(y, \hat{y}_{L2R}) + \mathcal{L}_{ce}(y, \hat{y}_{R2L}) + \lambda * \mathcal{D}(p_{L2R} || p_{R2L}), \quad (2)$$

where  $\lambda$  is the weight assigned to the consistency loss,  $p_{L2R}$  and  $p_{R2L}$  are the associated confidence/softmax vectors assigned by the model for  $L2R$  and  $R2L$  sentence pairs, and  $\mathcal{D}$  is one of the following:

1.  $KL(p||q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}$
2.  $JS(p||q) = \frac{1}{2}KL(p||m) + \frac{1}{2}KL(q||m)$ ,

Here  $p, q$  are probability distributions and  $m = \frac{1}{2}(p + q)$ . Minimizing divergences between two distributions brings them closer to each other.

## 4 Experimental Setup

### 4.1 Datasets

We experiment with 5 standard datasets from the GLUE benchmark (Wang et al., 2019) as well as the

PAWS dataset (Zhang et al., 2019)<sup>3</sup>. We categorize them under the following headings:

**A. For Symmetric Tasks:** (i) **QQP:** Quora Question Pairs (Iyer et al., 2017) data set contains pairs of questions marked with either 1 (paraphrases) or 0 (not paraphrases).

(ii) **PAWS:** Paraphrase Adv. from Word Scrambling (Zhang et al., 2019), contains human labeled sentence pairs annotated in line with QQP. The uniqueness about this dataset is the creation procedure which involves back-translation and word swapping. (iii) **MRPC:** Microsoft Research Paraphrase Corpus (Dolan and Brockett, 2005) comprises human annotated sentence pairs collected from newswire articles.

**B. For Single Input Task:** (i) **SST2:** Stanford Sentiment Treebank (Socher et al., 2013). This is a collection of human-annotated movie reviews. We work with the standard two class setting where the annotations have opposite polarities (1 for positive sentiment and 0 otherwise).

**C. For Non-symmetric tasks:** (i) **QNLI:** Natural Language Inference dataset constructed from SQuAD (Rajpurkar et al., 2016) related to a two-class classification problem to determine if the premise entails a hypothesis or not.

(ii) **RTE:** Recognizing Textual Entailment (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009) Corpus is a combination of multiple RTE datasets containing one of two labels (1 for entailment and 0 for non-entailment).

### 4.2 Evaluation

We analyse the results of the traditional objective as well as our approach on BERT-BASE and ROBERTA-BASE across four different seeds under the following categories:

1. **Prediction Consistency:** This evaluation is done only for the symmetric task. Score =  $\frac{\mathbb{1}_{(l_{L2R}=l_{R2L})}}{(\# \text{ of } L2R \text{ Samples})} * 100$ , where  $l_{L2R}, l_{R2L}$  denote labels for  $L2R$  and  $R2L$ , respectively. Note that this is not related to the ground truth labels.
2. **Confidence Consistency:** We perform these evaluations specifically for symmetric task.

<sup>3</sup>Since the test split of these datasets is not available in the GLUE benchmark (Wang et al., 2019), we use splits as given in Table 1. The validation dataset is kept as original and the new train and test sets are created by randomly splitting initial train data into train and test sets.

(A) <i>L2R</i> and <i>R2L</i> Prediction Consistency Mean $\pm$ stddev (Section 4.2: Evaluation [1])				(B) <i>L2R</i> and <i>R2L</i> Confidence Consistency Pearson Correlation [MSE * 1000] (Section 4.2: Evaluation [2])		
Models	QQP	PAWS	MRPC	QQP	PAWS	MRPC
BERT-BASE	96.6 $\pm$ 0.15	96.0 $\pm$ 0.54	91.1 $\pm$ 1.41	98.2 [5.89]	96.5 [14.2]	92.7 [17.0]
BERT-BASE W/ KL	<b>99.3 <math>\pm</math> 0.02</b>	<b>98.1 <math>\pm</math> 0.12</b>	<b>97.7 <math>\pm</math> 0.82</b>	99.9 [0.12]	99.6 [0.5]	99.5 [0.3]
BERT-BASE W/ JS	<b>98.9 <math>\pm</math> 0.05</b>	<b>98.1 <math>\pm</math> 0.22</b>	<b>96.9 <math>\pm</math> 0.93</b>	<u>99.8 [0.48]</u>	<u>99.3 [1.9]</u>	<u>99.0 [1.1]</u>
ROBERTA-BASE	97.0 $\pm$ 0.14	96.7 $\pm$ 0.25	91.5 $\pm$ 0.22	98.3 [5.90]	97.4 [10.8]	94.1 [16.3]
ROBERTA-BASE W/ KL	<b>99.3 <math>\pm</math> 0.03</b>	<b>98.9 <math>\pm</math> 0.11</b>	<b>97.4 <math>\pm</math> 0.78</b>	99.3 [0.10]	99.7 [0.4]	99.5 [0.3]
ROBERTA-BASE W/ JS	<b>99.1 <math>\pm</math> 0.05</b>	<b>98.7 <math>\pm</math> 0.23</b>	<b>96.7 <math>\pm</math> 1.11</b>	<u>99.8 [0.40]</u>	<u>99.6 [1.5]</u>	<u>99.0 [1.3]</u>

(C) Classification Performance Metrics (Section 4.2: Evaluation [3])						
Models	QQP (Acc/F1)	PAWS (Acc/F1)	MRPC (Acc/F1)	SST2 (Acc)	QNLI (Acc)	RTE (Acc)
BERT-BASE	89.5 / 85.7	91.1 / 90.1	78.3 / 82.7	94.0 $\pm$ 0.10	87.9 $\pm$ 0.13	63.0 $\pm$ 1.33
BERT-BASE W/ KL	87.1 / 82.3	88.0 / 86.8	73.0 / 80.7	94.1 $\pm$ 0.20	71.2 $\pm$ 4.15	51.6 $\pm$ 1.50
BERT-BASE W/ JS	89.7 / 86.0	90.5 / 89.5	76.6 / 82.6	94.2 $\pm$ 0.42	74.5 $\pm$ 0.80	50.2 $\pm$ 16.90
ROBERTA-BASE	90.2 / 87.2	92.6 / 91.7	82.4 / 86.0	94.4 $\pm$ 0.39	89.9 $\pm$ 0.47	70.6 $\pm$ 2.35
ROBERTA-BASE W/ KL	87.2 / 82.7	91.5 / 90.5	74.7 / 81.0	94.5 $\pm$ 0.36	85.3 $\pm$ 1.62	58.7 $\pm$ 5.40
ROBERTA-BASE W/ JS	90.0 / 86.6	92.3 / 91.6	79.2 / 84.9	95.1 $\pm$ 0.12	86.8 $\pm$ 1.51	61.4 $\pm$ 1.06

Table 2: **Parts (A) & (B):** *L2R* and *R2L* Prediction and Confidence Consistency. **Part (C) Classification Metrics.** (\*-BASE) indicate 🤖, (\*- W/ \*) indicate 🤖. Higher Accuracy, Higher Pearson Correlation and lower MSE are better. Numbers in **bold** are statistically significant. Underlined numbers are better on average than baselines. Please refer to Section 5.1 for a discussion.

This is to analyze how aligned are the confidence (softmax output associated with label 1) predicted by the model for *L2R* and *R2L* setting. The metrics used are the pearson correlation (scaled by 100) and the mean squared error (MSE - scaled by 1000) between the two confidence scores of the test data.

- Standard Classification Metrics:** These are task-specific metrics (accuracy/F1) used in the standard GLUE tasks (Wang et al., 2019)

### 4.3 Implementation Details

To fine-tune the model for symmetric classification tasks, we club together three paraphrase detection datasets: (a) QQP, (b) PAWS, and (c) MRPC. To make sure that all the models see the same data, we augment the dataset with its reverse samples during training. The model is then trained by passing the [CLSPara] (Section 3.2) representation through a low-capacity classifier, and optimized using Equation 1 for baseline models and Equation 2 for the consistency inducing models (Ours). We then use these models to conduct two sets of evaluations. We first evaluate the paraphrase detection results on QQP, PAWS, and MRPC individually. We then take the fine-tuned model obtained above and additionally fine-tune ([CLS] or <s> token) on the single input task (SST-2) and non-symmetric tasks (QNLI, RTE).

We use the hugging-face library (Wolf et al., 2020) for tokenizing the input, and the pytorch-lightning framework (Falcon et al., 2019) for loading the pre-

trained models and fine-tuning them. We optimize the objective using the AdamW (Loshchilov and Hutter, 2019) optimizer with a learning rate of  $2e-5$  (obtained through hyperparameter tuning { $2e-4$ ,  $2e-5$ ,  $4e-5$ ,  $2e-6$ }). Since the input contains an additional token [CLSPara], we extend the tokenizer vocabulary for each of the models. Each model was fine-tuned on a single Nvidia 1080Ti GPU (12 GB) for a maximum of 3 epochs ( $\approx$  6hrs/experiment). In case of BERT (Devlin et al., 2019), we use the bert-base-cased model while for RoBERTa (Liu et al., 2020), we use the RoBERTa-base model. For training stability, we perform lambda-annealing i.e., increase the  $\lambda$  parameter from 0.0 to 100.0 as the training progresses. This ensures that the model has developed the capability to classify the sentence pairs with some degree of correctness before making it adhere to the appropriate symmetric confidence scores. We also experimented with fixed  $\lambda$ , but the resultant models were slow to converge ( $\approx$  15 epochs).

## 5 Results

Our experiments address three questions:

- Q1.** What are the shortcomings of the current objective function for symmetric classification tasks? (Section 1, Section 5.2)
- Q2.** Does adding the consistency loss alleviate the inconsistency problem? (Section 5.1)
- Q3.** Can consistency-based fine-tuning improve other downstream tasks? (Section 5.1)

## 5.1 Quantitative Analysis

Table 2 presents our results. **Parts (A) & (B)** compare *L2R* and *R2L* models in terms of prediction consistency and confidence consistency. Models trained with the consistency loss (indicated by *W/\**) assign more similar predictions (indicated by higher scores in (A)) and confidence scores (indicated by higher correlation in (B)) as compared to the base model (indicated by *-BASE*), for both the base models (BERT-BASE/ROBERTA-BASE) and all symmetric test data sets (QQP, PAWS, MRPC). Moreover, the MSE (indicated within square brackets in part (B)) with consistency training is an order-of-magnitude smaller than without it. The improvements in part (A) are statistically significant at significance level ( $\alpha$ ) of 0.01 according to McNemar’s statistical test (Dror et al., 2018).

**Part (C)** shows the results on **downstream fine-tuning**. Our models (indicated by *W/\**) do not compromise significantly (statistically evaluated) on the classification metrics for QQP, PAWS, and MRPC (F1/accuracy). The consistency loss does not change the accuracy scores of single sentence input tasks (SST-2), but affects the non-symmetric tasks (QNLI, RTE) negatively. This seems natural since the final objective of both the tasks is quite different and, in many cases, uncorrelated or negatively correlated. Incorporating consistency loss before fine-tuning on non-symmetric tasks (such as entailment) should, therefore, be avoided.

**Limitations:** Our goal is to increase the reliability (measured in terms of confidence scores) of the model and not specifically target classification performance metrics like accuracy and F1. Cases where they increase, can only partially be attributed to a stricter consistency constraint.

## 5.2 Qualitative Analysis

We sample 30 instances that were assigned opposite labels for *L2R* and *R2L* by the BERT-BASE models (majority voting) for QQP, MRPC and PAWS. An evaluator with NLP expertise analysed these examples and grouped them into recall error types. We then check the predictions for the same set of instances from BERT + JS (recall). Counts for these error types (defined in Section 7.1) are shown in Table 3. Out of those 30 examples for QQP, MRPC and PAWS, 26, 26 and 23 respectively get corrected by 🤖. In general, the numbers reduce for all error types.

Error type	🤖	👤
<b>QQP</b>		
Different expected answer	4	0
Different answer type + Additional details	8	1
Different answer type + Additional details + Pronoun change	1	0
Additional details and/or pronoun change	17	3
<b>MRPC</b>		
Additional details missing	13	2
Reordering of phrases	3	0
Named entities and pronouns	6	1
Focus of sentences is different	6	0
Synonyms	2	1
<b>PAWS</b>		
Phrases are changed	10	4
Nouns/adjectives are changed	12	1
Nouns/adjectives and phrases are changed	4	0
Named entities are changed	3	1
Names entities and nouns/adjectives are changed	1	1

Table 3: Recall errors in QQP, MRPC & PAWS: BERT (🤖) and BERT with JS (👤). Please refer to Section 5.2.

## 6 Conclusion

In this paper, we proposed an additional objective: consistency loss between *L2R* and *R2L* predictions so as to alleviate the problem of input order-sensitive inconsistency in the case of symmetric classification tasks. For three symmetric classification tasks, our proposed solution, BERT-with-consistency-loss, results in an improved consistency in terms of Pearson’s correlation and MSE. As expected, consistency loss results in a drop in the performance of non-symmetric classification tasks such as QNLI and RTE. Surprisingly, using KL divergence results in marginally higher *consistency* than the JS counterpart. We leave this analysis for future work. Our qualitative analysis shows that all error types, including change in phrases or addition/deletion of details are reduced when the consistency loss is incorporated.

While consistency loss ensures that the predicted labels are the same even if the order of inputs is swapped, it can be adapted in the future to ensure expected outputs for anti-symmetric classification tasks (where  $\mathbb{P}(L2R) = 1 - \mathbb{P}(R2L)$ ) like next and previous sentence prediction, where reordering the inputs must result in an opposite predicted label. In addition, the proposed method can be applied to evaluate paraphrase generation models (Kumar et al., 2019, 2020) as well. In order to validate that paraphrasing models are indeed generating semantically similar outputs, BERT-with-consistency-loss can be used to either evaluate and filter out incorrect generations or be used as an objective to train learned metrics like BLEURT (Sellam et al., 2020).

## Ethical Considerations

The primary aim of this work is to highlight the inconsistency in labels and confidence scores of generated by standard pre-trained models for symmetric classification tasks. To mitigate the aforementioned inconsistency, we propose a loss function that incorporates divergence between outputs when the input order is swapped. We do not anticipate any additional ethical issues being introduced by our loss function as compared to the original standard pre-trained models, specifically BERT and RoBERTa. All the datasets used in our experiments are subset of the datasets from previously published papers, and to the best of our knowledge, do not have any attached privacy or ethical issues. That being said, further efforts should be made to study the inherent biases encoded in the pre-trained language models and the datasets.

## References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. [Neural machine translation by jointly learning to align and translate](#). In *International Conference on Learning Representations*.
- Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognizing textual entailment challenge. In *Proceedings of the second PASCAL challenges workshop on recognising textual entailment*, volume 6, pages 6–4. Venice.
- Luisa Bentivogli, Peter Clark, Ido Dagan, and Danilo Giampiccolo. 2009. The fifth pascal recognizing textual entailment challenge. In *TAC*.
- Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.
- 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, Minneapolis, Minnesota. Association for Computational Linguistics.
- William B Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.
- Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. [The hitchhiker’s guide to testing statistical significance in natural language processing](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.
- William Falcon et al. 2019. Pytorch lightning. *GitHub*. Note: <https://github.com/PyTorchLightning/pytorch-lightning>, 3:6.
- Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics.
- Yufang Huang, Wentao Zhu, Deyi Xiong, Yiye Zhang, Changjian Hu, and Feiyu Xu. 2020. [Cycle-consistent adversarial autoencoders for unsupervised text style transfer](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2213–2223, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Shankar Iyer, Nikhil Dandekar, and Kornel Csernai. 2017. [First quora dataset release: Question pairs](#).
- Ashutosh Kumar, Kabir Ahuja, Raghuram Vadapalli, and Partha Talukdar. 2020. Syntax-guided controlled generation of paraphrases. *Transactions of the Association for Computational Linguistics*, 8:330–345.
- Ashutosh Kumar, Satwik Bhattamishra, Manik Bhandari, and Partha Talukdar. 2019. [Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation](#). 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 3609–3619, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Ro{bert}a: A robustly optimized {bert} pretraining approach](#).
- Ilya Loshchilov and Frank Hutter. 2019. [Decoupled weight decay regularization](#). In *International Conference on Learning Representations*.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- 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, Hong Kong, China. Association for Computational Linguistics.
- Paul Rubenstein, Olivier Bousquet, Josip Djolonga, Carlos Riquelme, and Ilya O Tolstikhin. 2019. Practical and consistent estimation of f-divergences. *Advances in Neural Information Processing Systems*, 32:4070–4080.
- Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. **BLEURT: Learning robust metrics for text generation**. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.
- Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021. **Augmented SBERT: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks**. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 296–310, Online. Association for Computational Linguistics.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. **Attention is all you need**. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In the Proceedings of ICLR.
- Benyou Wang, Lifeng Shang, Christina Lioma, Xin Jiang, Hao Yang, Qun Liu, and Jakob Grue Simonsen. 2021. **On position embeddings in {bert}**. In *International Conference on Learning Representations*.
- Yu-An Wang and Yun-Nung Chen. 2020. **What do position embeddings learn? an empirical study of pre-trained language model positional encoding**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6840–6849, Online. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. **Transformers: State-of-the-art natural language processing**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Yuan Zhang, Jason Baldridge, and Luheng He. 2019. **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, Minneapolis, Minnesota. Association for Computational Linguistics.
- Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*.

## 7 Appendix

### 7.1 Recall Error Types in Qualitative Analysis

The qualitative analysis compares types of errors with and without consistency loss. The recall error types can be described as follows:

#### A. QQP:

- 1. Different expected answer:** This error is said to occur in the case of QQP when the two input questions have a different expected answer. An example of such a pair is: ‘*Is consciousness possible without self-awareness?*’ and ‘*Is self-awareness possible without consciousness?*’. The two questions are essentially complements of each other.
- 2. Different answer type + Additional details:** This error is said to occur when one of the inputs is structured in a way that the answer would solicit additional details. For example, the input pair ‘*How do I structure a big PHP project?*’ and ‘*How do I build a perfect PHP project?*’ are similar - but nuances between ‘structuring’ and ‘building’ a project may result in different answers.
- 3. Additional details and/or pronoun change:** The input pair ‘*What are the best ways to get thick and wavy hair?*’ and ‘*How can I get thick, wavy hair (as a guy)?*’ is similar - although the latter uses the first-person proverb.

Dataset	Example pair	True label	L2R Label	R2L Label
MRPC	(1) Shares in Wal-Mart closed at \$ 58.28 , up 16 cents , in Tuesday trading on the New York Stock Exchange. (2) Wal-Mart shares rose 16 cents to close at \$ 58.28 on the New York Stock Exchange.	1	0	1
	(1) Darren Dopp , a Spitzer spokesman , declined to comment late Thursday. (2) John Heine , a spokesman for the commission in Washington , declined to comment on Mr. Spitzer 's criticism.	0	0	1
QQP	(1) How do I retrieve my deleted history from Google chrome? (2) Can history be retrieved after deleting Google chrome?	1	0	1
	(1) Is consciousness possible without self-awareness? (2) Is self-awareness possible without consciousness?	0	1	0
PAWS	(1) This iteration is larger and has a smaller storage capacity than its previous versions. (2) This iteration is smaller and has a greater storage capacity than its previous versions	0	0	1
	(1) To get there , take Marine Drive west from the Lions Gate Bridge past Horseshoe Bay to Lighthouse Park and then continue on to 7100 Block Marine Drive. (2) To get there , take the Marine Drive from the Lions Gate Bridge to the west , past the Horseshoe Bay , Lighthouse Park and continue on to the 7100 Marine Drive block.	1	1	0

Table 4: Sample pairs which are classified differently by the fine-tuned model based on their input order in the standard classification setting in each of the paraphrase dataset. Please refer Section 1, Section 3.2 for details.

## B. MRPC:

- Additional details missing:** One of the inputs contains information (*i.e.*, details) that are not present in the other input. For example, ‘*The caretaker, identified by church officials as Jorge Manzon, was believed to be among the nine missing - some of them children*’ contains the number of missing persons that are not present in ‘*The caretaker, identified by church officials as Jorge Monzon, was believed to be among the missing, who are presumed dead*’.
- Reordering of phrases:** The two inputs contain the same information although the information may be represented using different phrasal structures. For example, ‘*Shares in Wal-Mart closed at \$ 58.28 , up 16 cents , in Tuesday trading on the New York Stock Exchange.*’ conveys the same information as ‘*Wal-Mart shares rose 16 cents to close at \$ 58.28 on the New York Stock Exchange.*’ The former uses passive voice while the latter uses ‘shares’ as the main verb.
- Named entities and pronouns:** One input replaces entities with pronouns, as in the case of ‘*The bonds traded to below 60 percent of face value earlier this year*’ and ‘*They traded down early this year to 60 percent of face*

*value on fears Aquila may default .*’

- Focus of sentences is different:** While information in one input is subsumed by the other, the latter might focus on a broader context. For example, ‘*A power cut in New York in 1977 left 9 million people without electricity for up to 25 hours*’ is implied in the sentence ‘*The outage resurrected memories of other massive power blackouts , including one in 1977 that left about 9 million people without electricity for 25 hours .*’ However, the latter describes a resurrection of memories of the event in 1977.
- Synonyms:** One or more words in an input may be replaced by its synonyms in the other input. For example, ‘*In 2001 , the number of death row inmates nationally fell for the first time in a generation*’ can be converted to ‘*In 2001 , the number of people on death row dropped for the first time in a decade.*’ by replacing the word ‘fell’ with ‘dropped’.

## C. PAWS

- Nouns/adjectives are changed:** In the case of these errors, adjectives are replaced. An example pair is ‘*This iteration is larger and has a smaller storage capacity than its previous versions*’ and ‘*This iteration is smaller*



*and has a greater storage capacity than its previous versions .’*

2. **Named entities are changed:** This refers to pairs where named entities (locations/people) are different. An example is the pair ‘*When Mexico was within Los Angeles , Botello was chief of staff for Mexican General Ramirez y Sesma . His two brothers also married daughters of the general’* and ‘*When Los Angeles was within Mexico , Botello was Chief of Staff of the Mexican General Ramirez y Sesma , his two brothers also married the general ’s daughters .’*