measure	% on the test set			Best TH
	prec	rec	F1	
NGRAM(n=1)*	77.8	82.2	79.9	0.3
NGRAM(n=2)*	77.8	82.2	79.9	0.3
NGRAM(n=3)	79.9	72.5	76.1	0.3
NGRAM(n=4)	77.8	82.2	79.9	0.3
NGRAM(n=5)	77.8	82.2	79.9	0.3
TOKENstring*	83.7	73.1	78.1	0.2
TOKENsyn	77.1	71.5	74.2	0.1
W2V	79.7	74.5	77.0	0.8
GLOVE	73.5	81.2	77.1	0.95
BERTword*	78.5	**87.0**	**82.5**	0.75
BERTcls	81.9	67.9	74.3	0.9
SBERTbert	75.2	90.8	82.3	0.6
SBERTalbert	82.9	70.7	76.9	0.35
SBERTmini*	78.4	85.2	81.6	0.6
BERTprec*	86.5	72.9	79.1	0.9
BERTrec*	83.5	74.9	80.4	0.9
BERTf1	**86.8**	74.9	80.4	0.9

Table 2: Alignment results for the Uni-directional Best Match strategy across all similarity measures. TH is the threshold value, selected on the development set based on the f1 value for each measure. The asterisk * marks the metrics that outperforms NGRAM baseline (n=3) with $p \leq 0.05$.

measure	% on the test set			Best TH
	prec	rec	F1	
NGRAM(n=1)	80.5	81.8	81.1	0.3
NGRAM(n=2)	80.5	81.8	81.1	0.3
NGRAM(n=3)	78.9	87.0	82.7	0.1
NGRAM(n=4)	80.5	81.8	81.1	0.3
NGRAM(n=5)	80.5	81.8	81.1	0.3
TOKENstring	84.7	73.1	78.5	0.2
TOKENsyn	78.6	81.8	80.2	0.05
W2V	81.1	87.6	84.2	0.6
GLOVE	79.7	78.0	78.8	0.95
BERTword	82.3	86.4	84.3	0.75
BERTcls	**86.2**	66.5	75.1	0.9
SBERTbert	79.1	88.6	83.6	0.6
SBERTalbert	80.6	89.8	84.9	0.25
SBERTmini*	80.7	90.2	**85.1**	0.25
BERTprec	80.9	88.2	84.4	0.85
BERTrec	79.7	88.2	83.7	0.85
BERTf1	79.9	**90.8**	85.0	0.9

Table 3: Alignment results for the Bi-directional Best Match strategy across all similarity measures. TH is the threshold value, selected on the development set based on the F1 value for each measure. The asterisk * marks the metrics that outperforms NGRAM baseline (n=3) with $p \leq 0.05$.

original work. We do not test our implementation on the original data adopted by them, as they only used human evaluation, without indicating which dataset was used for evaluation. Therefore, directly verifying our implementation with their results is impossible.

When experimenting with various search mechanisms, we additionally impose similarity score thresholding, which filters out those obtained 1-1 sentence pairs with their similarities below the specified threshold. The threshold value is selected for each similarity measure separately, based on the development set results.

For the approach of adopting [CLS] for sentence representation, we use a pretrained BERT-base model (Devlin et al., 2019). For the Sentence-BERT approach, we test three different pretrained versions released by an open resource[3]: BERT (Devlin et al., 2019; abbreviated as SBERTbert), ALBERT-mini (Lan et al., 2020; abbreviated as SBERTalbert), and MiniLM (Wang et al., 2020; abbreviated as SBERTmini). Among them, SBERTbert is trained with various *Natural Language Inference* data sets; in contrast, the last two versions are trained on various paraphrasing datasets [4]. The pre-trained model used for calculating the BERTScore is ROBERTA-Large (Liu et al., 2019).[5]

3.4 Various Experiments

We measure precision, recall, and F1-score for the two alignment strategies with various similarity measures. Furthermore, we use the McNemar test (Dietterich, 1998) to check if a given configuration (i.e., the adopted search mechanism and the specified similarity measure) yields significantly different results from the baseline (taking $p \leq 0.05$ as the significance test threshold).

We test the following measures: (A) **String-based** similarities: including character ngram similarity with *n* from 1 to 5 (NGRAM), and token overlap similarity calculated with either token strings (TOKENstring) or token synonyms (TOKENsyn); (B) **Embedding-based** similarities: (1) word embedding-based similarities calculated with word2vec (W2V), Glove (GLOVE) and BERTbase (BERTword) embeddings; (2) sentence embedding-based similarity: (i) using [CLS] token yielded by BERTbase model (BERTcls), and (ii)

[3] https://huggingface.co/sentence-transformers
[4] The list of specific datasets used was not published by the open-source authors.
[5] https://github.com/Tiiiger/bert_score