Sentence Level or Token Level Features for Automatic Short Answer Grading?: Use Both

Swarnadeep Saha, Tejas I. Dhamecha, Smit Marvaniya, Renuka Sindhgatta, and Bikram Sengupta

IBM Research-India
{swarnads,tidhamecha,smarvani,renuka.sr,bsengupt}@in.ibm.com

Abstract. Automatic short answer grading for Intelligent Tutoring Systems has attracted much attention of the researchers over the years. While the traditional techniques for short answer grading are rooted in statistical learning and hand-crafted features, recent research has explored sentence embedding based techniques. We observe that sentence embedding techniques, while being effective for grading in-domain student answers, may not be best suited for out-of-domain answers. Further, sentence embeddings can be affected by non-sentential answers (answers given in the context of the question). On the other hand, token level hand-crafted features can be fairly domain independent and are less affected by non-sentential forms. We propose a novel feature encoding based on partial similarities of tokens (Histogram of Partial Similarities or HoPS), its extension to part-of-speech tags (HoPSTags) and question type information. On combining the proposed features with sentence embedding based features, we are able to further improve the grading performance. Our final model achieves better or competitive results in experimental evaluation on multiple benchmarking datasets and a large scale industry dataset.

Keywords: Short Answer Grading, Sentence Embeddings, Histogram of Partial Similarity, Supervised Learning, Feature Fusion

1 Introduction

Short answer grading, an integral part of Intelligent Tutoring Systems, is positioned as a research problem at the intersection of natural language understanding and its application to educational technologies. Formally, the problem is to grade a student answer in the context of a question and its reference answer(s). The grades are either discrete or bounded real numbers. Thus, traditionally, the short answer grading problem is often modeled as a classification or regression task.

While early works on automatic short answer grading [17, 21, 25] used manually generated patterns from reference answers, recently Ramachandran et al. [23] proposed a method to automate the generation of patterns. However, patterns do not scale well across domains. They are not a good fit for grading non-sentential

Table 1: An illustrative example showing non-sentential and well-formed student answers.

Question	**Reference Answer**	**Student Answer 1 (Non-sentential)**	**Student Answer 2 (Well-formed)**
What is meant by monotonically increasing functions?	Monotonically increasing functions are the ones that are entirely not decreasing.	Entirely not decreasing.	Monotonically increasing functions are the ones that are either entirely increasing or remain constant.

student answers as they violate the structural patterns of the corresponding reference answers. The *non-sentential* answers, also called fragments [19], lack a complete sentential form but whose meaning can be inferred from the question context.

While some research efforts have proposed dedicated techniques for short answer grading, others view this problem as a specific application of generic natural language understanding tasks of textual entailment or semantic textual similarity. Traditionally, all machine learning based techniques have revolved around hand-crafted features. It is only recently that techniques have been proposed to utilize deep learning based approaches for short answer grading.

Hand-crafted Features

Mohler et al. [18] proposed a method involving graph alignment and lexical semantic similarity features. Heilman and Madnani [9] proposed a short answer scoring method that uses stacking and domain adaptation techniques. Jimenez et al. [10] proposed a 42 dimensional soft cardinality based feature representation for student answer analysis. The abstraction of the feature representation makes it more suitable for *unseen domain* scenarios. In an ensemble approach, Ott et al. [22] learn a meta-classifier by combining three different grade prediction systems. In one of the recent works, Sultan et al. [26] proposed a method for short answer grading in a feature ensemble approach involving text alignment, semantic similarity, question demoting, term weighting, and length ratios.

Overall, the hand-crafted features often rely on dependency or constituency parsers to encode the structural as well as semantic information of the student answers and the reference answers. However, it becomes restrictive in dialog-based tutoring systems, as the student answers can be non-sentential. Additionally, dependency parsers are slow and are not suitable for deployment in real-world systems.

Apart from these task-specific approaches, there have been numerous research efforts to model short answer grading as problems of textual entailment [5,12,21] or textual similarity [1–3, 13].

Deep Learning Approaches

Deep learning techniques, mostly using Recurrent Neural Networks (RNN) and their variants, particularly Long Short-Term Memory (LSTM) have achieved

state-of-the-art results in various natural language understanding tasks including textual similarity and textual entailment. A few of such efforts include an LSTM based approach [4], a Siamese LSTM model [20] and a Siamese bi-LSTM model using earth mover's distance [11]. Recently, LSTMs have also been used in the education community for detecting misconceptions from students responses [14]. Conneau et al. [7] proposed a method, termed as InferSent, for learning universal sentence representations using a bi-LSTM network with max pooling. They trained the network on the Stanford Natural Language Inference corpus [6] to generate sentence embeddings, which are shown to work well across various natural language understanding tasks.

These techniques either train an end-to-end deep neural network or learn an embedding network followed by training a classifier. The former requires a large labeled data to learn, which is often a limitation for short answer grading. The later (embedding learning techniques) can be problematic for non-sentential answers, which is a common occurrence in dialog-based tutoring systems. However, Conneau et al. [7] showed that their pre-trained sentence embeddings can be transferred to other NLP tasks without having to learn them. Thus, we use these embeddings to obtain sentence level features for short answer grading.

Limitations and Research Contributions

We observe that in dialog-based tutoring systems, the student answers are either 1) well-formed, or 2) non-sentential responses. Table 1 presents such an example. The second category of answers can derail techniques that depend on accurate parsing, non-contextual completeness, and grammatical structure of answers. Further, there is significant scope for fine-tuning hand-crafted features to make them suitable for *partial* answers in dialog-based tutors. We have also observed that hand-crafted features generalize better across domains, as compared to sentence embedding based approaches. To this end, we believe that using hand-crafted features in conjunction with sentence embedding features is necessary for improved short answer grading. In keeping with this goal, we make the following salient contributions.

- We develop novel token level features that are specifically tuned for understanding partially correct student answers. We call them Histogram of Partial Similarities (HoPS).
- Certain Part-of-Speech tags are more important than others for certain question types. Thus, we use question type information and combine HoPS per POS Tag of the expected answer tokens to further refine our features. To the best of our knowledge, using question types and POS tags as features for short answer grading is another novel contribution of our work.
- Our features are fast, easy to compute, and domain-independent.
- We combine token level features with sentence level features obtained using InferSent. The effectiveness of this expanded feature set is verified empirically across a variety of short answer grading tasks and datasets.

We also present comparable or better results than previously reported state-of-the-art results on SemEval-2013 task, Mohler et al. dataset, and a large-scale

industry dataset. Further, we showcase that our features work equally well on both in-domain and out-of-domain data across various classification and regression tasks in short answer grading. The following section provides a detailed description of the proposed features.

2 Proposed Features

We devise and combine two broad categories of features: 1) sentence level features and 2) token level features. The following two subsections discuss both the categories as well as the individual components of our token level features - Word Overlap (WO), Question Types (QT) and Histogram of Partial Similarity on POS tags (HoPSTag). Our proposed architecture combining sentence and token level features is shown in Fig. 1.

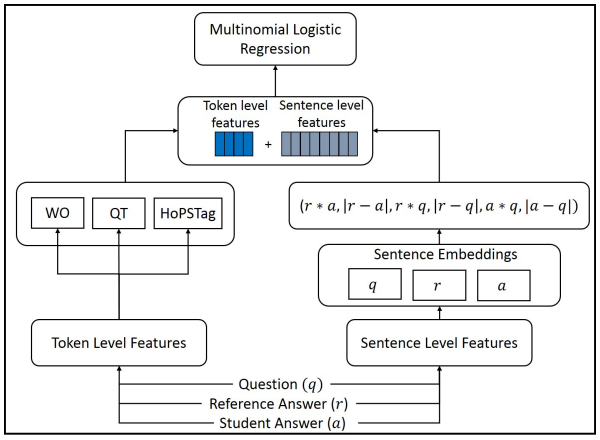

Fig. 1: Our proposed architecture that combines sentence and token level features. WO, QT and HoPSTag refer to word overlap, question type, and histogram of partial similarity on POS tags, respectively.

2.1 Sentence Level Features

For a triple *(question, reference answer, student answer)*, we first obtain sentence embeddings for the question (q), reference answer (r) and student answer (a) using InferSent [7][1]. Once the sentence embeddings are generated, we compute the sentence level features as:

$$S_{feat}(q, r, a) = (r * a, |r - a|, r * q, |r - q|, a * q, |a - q|) \quad (1)$$

where $r*a$ represents the element-wise multiplication of vectors r and a, while $|r-a|$ represents their absolute element-wise difference. Overall, this representation captures 1) the information gap between student and reference answer ($r*a$ and $|r-a|$), 2) the novel information expected in answer ($r*q$ and $|r-q|$), and 3)

[1] *https://github.com/facebookresearch/InferSent*

the novel information expressed in the student answer ($a * q$ and $|a - q|$). We find that incorporating the question embedding in the sentence level features is novel to our feature representation and is particularly useful in scenarios where the question at test time is already seen at train time. The following subsection describes the token level feature extraction process in detail.

2.2 Token Level Features

The idea of token level features largely stems from the fact that in a dialog-based tutoring scenario, students often only mention the important tokens. Such responses, although correct, are non-sentential and may not necessarily entail the reference answer. The preprocessing step for calculating token level features first involves stop words removal from the reference answer and the student answer. Subsequently, question demoting [18, 26] is performed that removes words from reference answer and student answer that are also present in the question. This leads to the generation of two bags of words, one for the reference answer and another for the student answer. The token level features are based on the following three key insights.

Word Overlap: We extract token based similarity features using the reference answer bag (RA) and the student answer bag (SA). The similarity features are calculated based on the number of word overlaps between the bags. A word w_i in reference answer bag is considered overlapping if 1) its score with student answer bag, as calculated in Eq. 2, is greater than a certain threshold δ, or 2) it is part of the Wordnet [16] synsets of any word from the student answer bag. $Cos(w_i, w_j)$ represents the cosine similarity between the word vectors of words w_i and w_j. Based on the number of word overlaps between the bags, we calculate three features - 1) Precision - word overlap count upon the number of words in the student answer bag, 2) Recall - word overlap count upon the number of words in the reference answer bag, and 3) Precision$\times$Recall.

$$Score(w_i, SA) = \max_{w_j \in SA} Cos(w_i, w_j), \text{ where } w_i \in RA \tag{2}$$

Histogram of Partial Similarity (HoPS): The objective of computing the partial similarity histogram is to capture the similarity pattern between the student answer and the reference answer. For each word w_i in reference answer bag, we compute the similarity score $Score(w_i, SA)$ with respect to student answer bag (SA) using Eq. 2. The similarity between two words is computed as the cosine similarity between their word embeddings. The similarity score for a word w_i with respect to the student answer SA is the maximum of the its similarities with all the tokens in SA. These similarity scores are partitioned into N bins. The bin index I for word w_i is computed as:

$$I(w_i) = min\left(\left\lfloor \frac{Score(w_i, SA) + 1}{h} \right\rfloor, N - 1\right) \text{ where } h = \frac{2}{N} \tag{3}$$

where indices are zero-indexed, and h is the width of the bin, and N is the number of bins. A histogram is created after binning each word in the reference answer bag. Our HoPS features are calculated by dividing each bin value by the size of the reference answer bag. This ensures that the HoPS features are invariant of the size of the reference answer bag. Unlike word overlap features, HoPS incorporates all words in the reference answer bag without thresholding on a similarity score, and thus helps model the class of partially correct student answers.

Histogram of Partial Similarity with POS Tags (HoPSTags) and Question Types: As an extension to HoPS, we segregate the histogram counts on the basis of various part of speech tags of the reference answer tokens. Specifically, for each of the 5 POS tags - Verb, Noun, Adjective, Adverb and Other, we create a histogram as before. The bin values of each histogram are divided by the number of tokens in the reference answer bag having the corresponding POS tag. We term this extended set of features as HoPSTag.

The utility of these features is understood best in the context of specific types of questions. Depending on the question type, reference answer tokens of certain POS tags become more important than others. For example, a factoid question like *'Who is the prime minister of India'* expects a noun (*Modi*) in the student answer. We find that short answer grading questions almost always belong to a fixed taxonomy of questions types, given by the set {*How, What, Why, Who, Which, When, Where, Whom*}. Thus, we generate 8 binary features, one for each question type depicting the presence of that question type. Table 2 shows the question type distributions of the SemEval-2013 dataset and the large scale dataset.

In Fig. 2, we show a HoPSTag feature representation averaged over all the partially correct student answers from SemEval-2013 5-way classification dataset. A significant number of reference answer tokens in the middle to high similarity bins indicates partially correct student answers. In a strict word overlap based setting, a large portion of these similarity values may not get incorporated; which we are able to preserve in HoPSTag features.

Table 2: Question type distributions of SemEval-2013 dataset and the large scale industry dataset.

	How	What	Why	Who	Which	When	Where	Whom	Other	Total
SemEval-2013 dataset										
Train	38	58	44	4	19	34	2	0	10	135
Unseen Answer	38	58	44	4	19	34	2	0	10	135
Unseen Question	4	8	4	1	1	4	0	0	1	15
Unseen Domain	10	23	12	0	6	15	3	0	0	46
Large-scale industry dataset										
Train and Test	172	278	13	13	6	19	2	2	0	483

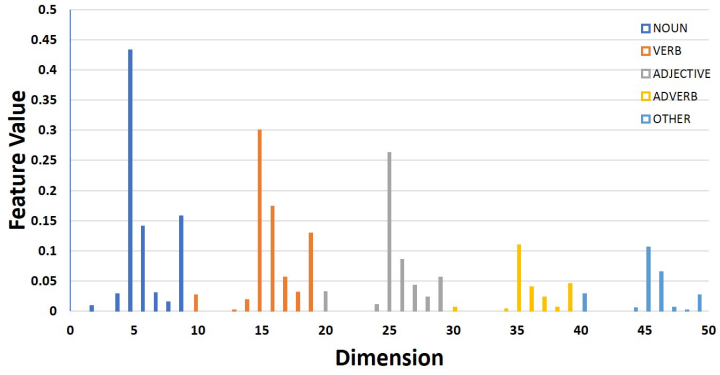

Fig. 2: A 50-dimensional average HoPSTag feature representation over all partially correct student answers from SemEval-2013 5-way classification subtask.

2.3 Combined Features

Our final feature representation is a combination of the sentence and token level features. Each sentence embedding is a S dimensional vector, making our sentence level feature size as $6S$ dimensional. Token level feature representation is a concatenation of 3 word overlap features, 8 binary question type features, and $5N$ HoPSTag features, where N is the number of bins. Thus, the final combined feature representation is $6S + 5N + 11$ dimensional.

3 Experiments

To evaluate the effectiveness of our proposed approach, we perform experiments on three datasets.

1. **Large Scale Industry Dataset:** The dataset consists of student answers collected for Psychology domain. The task is a 3-way classification task which requires predicting a student answer as `correct`, `partial`, or `incorrect`.
2. **SemEval-2013 [8] Dataset:** We use the SciEntsBank corpus of the SemEval-2013 dataset consisting of 197 questions in Science domain. The task involves three classification subtasks on unseen-answers (UA), unseen-question (UQ), and unseen-domain (UD) scenarios.
3. **Mohler et al. [18] Dataset:** This dataset is from Computer Science domain and consists of 80 undergraduate data structures questions. The task is a regression task and requires computing a real valued score for a student answer on a scale of 0 to 5.

Table 3 provides details pertaining to dataset sizes and experimental protocols. Further, we provide an ablation study to analyze the importance of individual

components of the proposed feature representation. For the large scale industry dataset and the SemEval-2013 dataset, the results are reported in terms of accuracy (Acc), macro-average F1 (M-F1), and weighted-F1 (W-F1). Since Mohler's task is a regression one, the performance is measured in terms of root mean square error (RMSE) and Pearson's correlation.

Table 3: Characteristics of the large scale industry dataset, SemEval-2013 dataset, and Mohler's dataset.

Dataset	**Responses**	**Protocol**			
		Train	**Test**		
Large scale industry dataset	16,458	12,317	4,141		
SemEval-2013 dataset [8]	10,804	4,969	UA	UQ	UD
			540	733	4,562
Mohler et al. dataset [18]	2,273 (12 assignments)	Leave-one-assignment-out			

In all our experiments, we use NLTK[2] for stop words removal and Scikit-learn[3] for the classifiers. We have used $N = 10$ as the number of bins, overlap threshold $\delta = 0.7$, and sentence embedding size $S = 4,096$. For the large scale industry dataset, we specifically use word vectors trained on a Psychology domain corpus from which the questions have been curated. For others, we use pre-trained word vectors [15]. Our 61 dimensional token level features are trained using a Random Forest classifier with 100 estimators. All reported results using Random Forests are averaged across 100 runs of the system. The sentence level features as well as its combination with token level features are trained using a Multinomial Logistic Regression classifier. The best parameters are chosen using 5-fold cross-validation over the training data. The regression task on Mohler et al. dataset is trained using a ridge regression model.

While reporting comparative results for these datasets, we also show experiments with five variants of the proposed features, including 1) only token level features (TF), 2) only sentence level features without question embedding (SF(-Q)), 3) only sentence level features with question embedding (SF(+Q)), 4) combination of token level and sentence level features without question embedding (TF+SF(-Q)), and 5) combination of token level and sentence level features with question embedding (TF+SF(+Q)).

3.1 Large Scale Industry Dataset

Our first experiment is on a first of its kind large scale industry dataset on Psychology domain. The test data has the same set of questions and reference answers as the training set. We compare our models against SEMILAR [24] and

[2] *http://www.nltk.org/*
[3] *http://scikit-learn.org/stable/*

Sultan et al. [26][4]. SEMILAR[5] is a semantic similarity toolkit, widely used in various core NLP tasks including paraphrase identification, question answering, and short answer grading. We choose to compare against Sultan et al. [26] as their model outperformed all existing models in short answer grading. Their system was trained using a Random Forest classifier with 500 estimators, as mentioned in their paper. Table 4 shows the comparative results of all the models. We list the salient observations below.

Table 4: Results of the proposed approach on large scale industry dataset.

	Acc	**M-F1**	**W-F1**
SEMILAR	0.3485	0.3367	0.3470
Sultan et al. [26]	0.5274	0.4722	0.5111
TF	0.5881	0.5419	0.5749
SF(-Q)	0.6184	0.5773	0.6066
SF(+Q)	0.6508	0.6177	0.6426
TF+SF(-Q)	0.6341	0.5958	0.6229
TF+SF(+Q)	**0.6636**	**0.6309**	**0.6558**

- Our token level features (TF) alone significantly outperforms SEMILAR and Sultan et al. [26]'s system in all the three metrics.
- The sentence level features (SF) outperform our token level features. Further, our results are significantly better when the question embedding is used for feature encoding (SF(+Q)). This, however, is unsurprising because the questions at test time were already seen at training time, thus enabling the classifier to capture the semantics of the question as well.
- Results improve further when we combine our token level features and sentence level features (TF+SF(+Q)), leading to an overall gain of 14 points in Weighted-F1 over the existing best system of Sultan et al. [26].

3.2 SemEval-2013 [8] Dataset

SemEval-2013 task on student answer analysis consists of three classification subtasks - 1) 2-way classification into `correct` and `incorrect` classes, 2) 3-way classification into `correct`, `incorrect` and `contradictory` classes, 3) 5-way classification into `correct`, `partially correct`, `contradictory`, `irrelevant` and `contradictory` classes. Each subtask comprises of three kinds of test data - Unseen Answers (UA), Unseen Questions (UQ) and Unseen Domains (UD). As shown in Table 3, the split sizes as well as the samples are exactly same in all the three subtasks. However, the labels change as the subtasks become more fine grained. For this dataset, we show comparative results with four systems including the top performing ones from the task leader board[6]. Particularly, we

[4] All experiments on Sultan et al. 's system were performed using their publicly available code at *https://github.com/ma-sultan/short-answer-grader*

[5] *http://www.semanticsimilarity.org/*

[6] https://docs.google.com/spreadsheets/d/1Xe3lCi9jnZQiZW97hBfkg0x4cI3oDfztZPhK3TGO_gw/pub?output=html#

present comparisons with CoMeT [22], ETS [9], SOFTCAR [10], and Sultan et al. [26]. The results are shown in Table 5. We summarize our key observations below.

Table 5: Results for all the classification subtasks of SemEval-2013 task.

(a) 2-way

	UA			UQ			UD		
	Acc	**M-F1**	**W-F1**	**Acc**	**M-F1**	**W-F1**	**Acc**	**M-F1**	**W-F1**
CoMeT	0.7740	**0.7680**	**0.7730**	0.6030	0.5790	0.5970	0.6760	0.6700	0.6770
ETS	**0.7760**	0.7620	0.7700	0.6330	0.6020	0.6220	0.6270	0.5430	0.5740
SOFTCAR	0.7240	0.7150	0.7220	**0.7450**	**0.7370**	**0.7450**	0.7110	0.7050	0.7120
Sultan et al. [26]	0.7087	0.6768	0.6907	0.7050	0.6786	0.6951	**0.7129**	**0.7038**	**0.7121**
TF	0.7451	0.7339	0.7414	0.6878	0.6517	0.6717	0.7097	0.7009	0.7091
SF(-Q)	0.7315	0.7250	0.7308	0.6753	0.6617	0.6738	0.6357	0.5909	0.6125
SF(+Q)	0.7481	0.7422	0.7476	0.6303	0.6204	0.6314	0.6324	0.6109	0.6253
TF+SF(-Q)	0.7796	0.7710	0.7771	**0.7490**	**0.7385**	**0.7478**	0.7087	0.6903	0.7023
TF+SF(+Q)	**0.7926**	**0.7858**	**0.7910**	0.7026	0.6850	0.6983	**0.7196**	**0.7089**	**0.7178**

(b) 3-way

	UA			UQ			UD		
	Acc	**M-F1**	**W-F1**	**Acc**	**M-F1**	**W-F1**	**Acc**	**M-F1**	**W-F1**
CoMeT	0.7130	0.6400	0.7070	0.5460	0.3800	0.5220	0.5790	0.4040	0.5500
ETS	**0.7200**	**0.6470**	**0.7080**	0.5830	0.3930	0.5370	0.5430	0.3330	0.4610
SOFTCAR	0.6590	0.5550	0.6470	**0.6520**	**0.4690**	**0.6340**	**0.6370**	**0.4860**	**0.6200**
Sultan et al. [26]	0.6042	0.4439	0.5696	0.6426	0.4550	0.6154	0.6269	0.4516	0.6033
TF	0.6489	0.5537	0.6385	0.6152	0.4239	0.5844	0.6325	0.4491	0.6084
SF(-Q)	0.6963	0.6408	0.6908	0.5484	0.4508	0.5594	0.5601	0.4217	0.5322
SF(+Q)	0.6907	0.6373	0.6860	0.5948	0.4541	0.5818	0.5686	0.4176	0.5468
TF+SF(-Q)	**0.7185**	**0.6662**	**0.7143**	0.6139	**0.4912**	0.6281	0.6324	**0.4794**	**0.6115**
TF+SF(+Q)	0.7185	0.6574	0.7112	**0.6535**	0.4890	**0.6362**	**0.6403**	0.4524	0.6107

(c) 5-way

	UA			UQ			UD		
	Acc	**M-F1**	**W-F1**	**Acc**	**M-F1**	**W-F1**	**Acc**	**M-F1**	**W-F1**
CoMeT	0.6000	0.4410	0.5980	0.4370	0.1610	0.2990	0.4210	0.1210	0.2520
ETS	**0.6430**	**0.4780**	**0.6400**	0.4320	0.2630	0.4110	0.4410	**0.3800**	0.4140
SOFTCAR	0.5440	0.3800	0.5370	**0.5250**	**0.3070**	**0.4920**	**0.5120**	0.3000	**0.4710**
Sultan et al. [26]	0.4898	0.3298	0.4875	0.4808	0.3020	0.4676	0.5065	0.3440	0.4847
TF	0.5523	0.3920	0.5510	0.4966	0.3628	0.4748	0.5194	0.2986	0.4822
SF(-Q)	0.6111	0.4603	0.6098	0.3943	0.3395	0.3973	0.4014	0.3229	0.4004
SF(+Q)	0.6074	0.4645	0.6089	0.4325	0.3127	0.3964	0.4213	0.3114	0.4064
TF+SF(-Q)	**0.6444**	**0.4808**	**0.6420**	0.5007	0.3168	**0.4881**	0.5088	**0.3574**	**0.4923**
TF+SF(+Q)	0.6296	0.4724	0.6303	**0.5061**	**0.3763**	0.4719	**0.5107**	0.3420	0.4862

– State-of-the-art techniques do not perform well across all kinds of test data. However, our domain-independent token level features have balanced per-

formance throughout, as shown by competitive results in 2-way, 3-way, and 5-way across UA, UQ and UD (See *TF* rows in the tables).
- The sentence level features, on the other hand, suffer from lack of domain adaptability, as shown by significantly poor results on UD (See *SF* rows).
- We see substantial improvement in all the metrics when we combine our token level and sentence level features (See *TF+SF* rows). Specifically, in UA, we achieve 5, 2, and 4 points improvement in 2-way, 3-way, and 5-way respectively. This again testifies the need for both kinds of features and also verifies our hypothesis.
- Our final model combining sentence and token level features (TF+SF) outperforms or produces competitive results as compared to all existing models. We find it even more creditable, since there is no single state-of-the-art model that has best performances across all the test sets.

3.3 Mohler et al. [18] Dataset

The dataset includes responses from 12 Computer Science assignments. The experimental protocol uses one assignment each for test and development set, and the remaining ten assignments as the train set. This procedure is repeated 12 times, once per assignment. Finally, the RMSE and Pearson's correlation are computed between the predicted scores and the ground truths across all the assignments. For this dataset, we compare results against four approaches, including Mohler et al. [18] and Sultan et al. [26]. The results are reported in Table 6. The key observations are listed below.

Table 6: Results on Mohler et al. [18] dataset.

	Pearson's r	**RMSE**
tf-idf†	0.327	1.022
Lesk†	0.45	1.05
Mohler et al. [18]	**0.518**	**0.978**
Sultan et al. [26]	**0.592**	**0.887**
TF	0.531	0.929
SF(-Q)	0.376	1.015
SF(+Q)	0.448	0.98
TF+SF(-Q)	0.542	0.921
TF+SF(+Q)	**0.570**	**0.902**

† Results are as reported in Mohler et al. [18].

- While sentence level features benefit from encoding the question information, the overall performance is still poorer compared to the token level features. This is largely down to the presence of a sizable number of factoid questions in the dataset.
- Compared to Mohler et al. [18], we report significant improvement. However, we report competitive results compared to Sultan et al. [26]. Note that Sultan et al. [26] relies on dependency parsing, which is computationally expensive and slow compared to the proposed features.
- Similar to experiments on the previous two datasets, the combination of token and sentence level features again shows significant improvement as compared to when used individually.

3.4 Ablation study of Token Level Features

In order to gain a detailed understanding of our proposed features, we perform an ablation study on the large scale industry dataset. We start with the simple 3 dimensional word overlap (WO) features, and cumulatively add HoPS, POS tag, HOPSTag, and question type (QT) features to finally obtain the 61 dimensional representation (WO+HoPSTag+QT). Table 7 shows the results of the ablation study on large scale industry dataset. We list the key observations.

Table 7: Ablation study of our Token Level features on large scale industry dataset.

	Dim	**Acc**	**M-F1**	**W-F1**
WO	3	0.4868	0.376	0.4398
WO+HoPS	13	0.5077	0.4728	0.5069
WO+POS tag	8	0.4957	0.4116	0.4658
WO+POS tag+QT	16	0.5147	0.4855	0.5183
WO+HoPSTag	53	0.5646	0.5153	0.5509
WO+HoPSTag+QT	61	**0.5881**	**0.5419**	**0.5749**

- The WO features do not capture the partial similarity between the student and the reference answers, as shown by very low macro-average F1.
- Encoding partial similarities using HoPS features significantly improves the performance, particularly in terms of macro-average F1.
- Further, computing HoPSTag features yield additional 4 points improvement in macro-average F1.
- Enriching the HoPSTag features with question type information yields another 3 points improvement. This validates the intuition that tokens of certain POS tags are important for certain question types.

4 Conclusion

In this work, we propose an approach combining token and sentence level features for short answer grading. By using the proposed features, we show that we can overcome the limited accuracy of token level features and also the domain dependence of sentence level features. Our feature representation is based on three key insights – 1) the partial similarities of tokens between the reference and the student answer, termed as HoPS, 2) its extension to POS tags, termed as HoPSTags, and 3) question type information. Empirical evaluation of the proposed approach across benchmarking datasets show better or competitive results as compared to state-of-the-art. This demonstrates the effectiveness and generalizability of our method. Overall, the results suggest that sentence and token level features encode some non-overlapping aspects of information. We believe that this observation will be helpful in devising better features in the broader domain of semantic textual similarity. This remains one of the key future directions of our work.

References

1. Agirre, E., Banea, C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W., Lopez-Gazpio, I., Maritxalar, M., Mihalcea, R., et al.: SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability. In: Proceedings of the International Workshop on Semantic Evaluation. pp. 252–263 (2015)
2. Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W.: * SEM 2013 shared task: Semantic textual similarity. In: Proceedings of the Joint Conference on Lexical and Computational Semantics. vol. 1, pp. 32–43 (2013)
3. Agirre, E., Diab, M., Cer, D., Gonzalez-Agirre, A.: Semeval-2012 task 6: A pilot on semantic textual similarity. In: Proceedings of the Joint Conference on Lexical and Computational Semantics. pp. 385–393 (2012)
4. Alikaniotis, D., Yannakoudakis, H., Rei, M.: Automatic text scoring using neural networks. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. vol. 1, pp. 715–725 (2016)
5. Bjerva, J., Bos, J., Van der Goot, R., Nissim, M.: The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity. In: Proceedings of the International Workshop on Semantic Evaluation. pp. 642–646 (2014)
6. Bowman, S.R., Angeli, G., Potts, C., Manning, C.D.: A large annotated corpus for learning natural language inference. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (2015)
7. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. pp. 670–680 (2017)
8. Dzikovska, M.O., Nielsen, R.D., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., Dang, H.T.: SemEval-2013 Task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In: Proceedings of the NAACL-HLT International Workshop on Semantic Evaluation. pp. 263–274 (2013)
9. Heilman, M., Madnani, N.: ETS: Domain adaptation and stacking for short answer scoring. In: Proceedings of the Joint Conference on Lexical and Computational Semantics. vol. 2, pp. 275–279 (2013)
10. Jimenez, S., Becerra, C., Gelbukh, A.: SOFTCARDINALITY: Hierarchical text overlap for student response analysis. In: Proceedings of the Joint Conference on Lexical and Computational Semantics. vol. 2, pp. 280–284 (2013)
11. Kumar, S., Chakrabarti, S., Roy, S.: Earth movers distance pooling over siamese lstms for automatic short answer grading. In: Proceedings of the International Joint Conference on Artificial Intelligence. pp. 2046–2052 (2017)
12. Levy, O., Zesch, T., Dagan, I., Gurevych, I.: Recognizing partial textual entailment. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics. vol. 2, pp. 451–455 (2013)
13. Lopez-Gazpio, I., Maritxalar, M., Gonzalez-Agirre, A., Rigau, G., Uria, L., Agirre, E.: Interpretable semantic textual similarity: Finding and explaining differences between sentences. Knowledge-Based Systems 119, 186–199 (2017)
14. Michalenko, J.J., Lan, A.S., Baraniuk, R.G.: D.TRUMP: data-mining textual responses to uncover misconception patterns. In: Proceedings of the Fourth ACM Conference on Learning @ Scale, L@S. pp. 245–248 (2017)

15. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the Advances in Neural Information Processing Systems. pp. 3111–3119 (2013)
16. Miller, G.A.: WordNet: a lexical database for English. Communications of the ACM 38(11), 39–41 (1995)
17. Mitchell, T., Russell, T., Broomhead, P., Aldridge, N.: Towards robust computerised marking of free-text responses. In: Proceedings of the International Computer Assisted Assessment Conference (2002)
18. Mohler, M., Bunescu, R.C., Mihalcea, R.: Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. pp. 752–762 (2011)
19. Morgan, J.: Sentence fragments and the notion sentence. Issues in linguistics: Papers in honor of Henry and Renée Kahane pp. 719–751 (1973)
20. Mueller, J., Thyagarajan, A.: Siamese recurrent architectures for learning sentence similarity. In: Proceedings of the Association for the Advancement of Artificial Intelligence. pp. 2786–2792 (2016)
21. Nielsen, R.D., Ward, W., Martin, J.H.: Recognizing entailment in intelligent tutoring systems. Natural Language Engineering 15(4), 479–501 (2009)
22. Ott, N., Ziai, R., Hahn, M., Meurers, D.: CoMeT: Integrating different levels of linguistic modeling for meaning assessment. In: Proceedings of the Joint Conference on Lexical and Computational Semantics. vol. 2, pp. 608–616 (2013)
23. Ramachandran, L., Cheng, J., Foltz, P.: Identifying patterns for short answer scoring using graph-based lexico-semantic text matching. In: Proceedings of the Workshop on Innovative Use of NLP for Building Educational Applications. pp. 97–106 (2015)
24. Rus, V., Lintean, M., Banjade, R., Niraula, N., Stefanescu, D.: Semilar: The semantic similarity toolkit. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics: System Demonstrations. pp. 163–168 (2013)
25. Sukkarieh, J.Z., Pulman, S.G., Raikes, N.: Auto-marking 2: An update on the UCLES-Oxford University research into using computational linguistics to score short, free text responses. International Association of Educational Assessment (2004)
26. Sultan, M.A., Salazar, C., Sumner, T.: Fast and easy short answer grading with high accuracy. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1070–1075 (2016)