The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Proposition Entailment in Educational Applications Using Deep Neural Networks

Florin Bulgarov, Rodney Nielsen
University of North Texas
1155 Union Circle, Denton, Texas, 76203, USA
FlorinBulgarov@my.unt.edu
Rodney.Nielsen@unt.edu

Abstract

The next generation of educational applications need to significantly improve the way feedback is offered to both teachers and students. Simply determining coarse-grained entailment relations between the teacher's reference answer as a whole and a student response will not be sufficient. A finer-grained analysis is needed to determine which aspects of the reference answer have been understood and which have not. To this end, we propose an approach that splits the reference answer into its constituent propositions and two methods for detecting entailment relations between each reference answer proposition and a student response. Both methods, one using hand-crafted features and an SVM and the other using word embeddings and deep neural networks, achieve significant improvements over a state-of-the-art system and two alternative approaches.

1 Introduction

Recent advancements in machine learning have started to put their mark on educational technology. Although the vast majority of the classrooms around the world look essentially the same as they have for several decades, many teachers and students have started to embrace the advantages that technology can bring to the learning process. This paper focuses on increasing the learning gains in classrooms through technology that enhances the analysis between the teacher's reference answer, a student's response, and the relations between them. We contribute by proposing two fine-grained approaches that predict entailment relations between a student's response and each proposition or clause from the teacher's answer. Both methods, one that uses neural networks with word embeddings and the other an SVM model with hand-crafted features, reach similar average F_1-scores, significantly outperforming a state-of-the-art system and two alternative approaches.

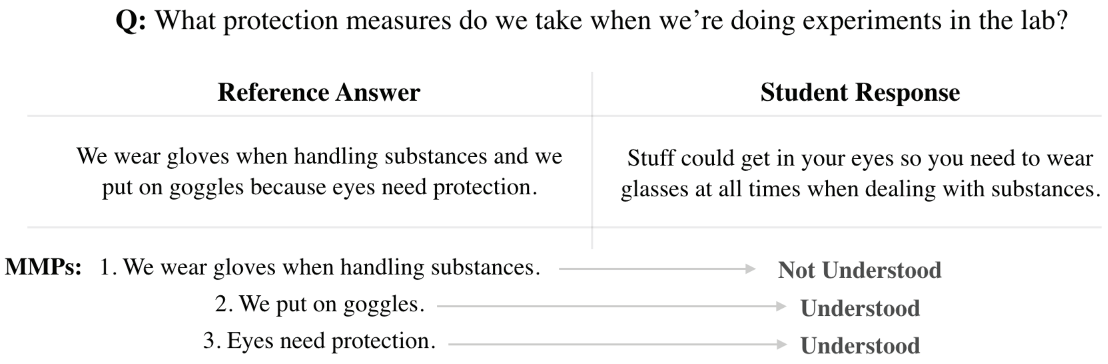
Intelligent educational applications haven't replaced humans, but they can definitely disrupt the way students have been acquiring knowledge. Several types of such systems have been developed in the recent years. Groups of researchers have come up with smart algorithms and applications that attempt to maximize the effectiveness of the learning process by improving the feedback given to students (Heffernan and Heffernan 2014), assessing students' understanding of a concept (Leacock and Chodorow 2003; Sukkarieh and Stoyanchev 2009), increasing classroom engagement (Paiva et al. 2014), facilitating self-guided learning and liberating instructors from doing repetitive tasks such as grading answers (Horbach, Palmer, and Pinkal 2013).

Among the first types of educational technology were the assessment systems (Mitchell et al. 2002; Leacock and Chodorow 2003; Nielsen, Ward, and Martin 2009; Sukkarieh and Stoyanchev 2009). Such systems usually quantify the similarity between a student response and a reference answer provided by the teacher or a content matter expert. The result is a score, or a grade, which is communicated to the student or the instructor.

Intelligent Tutoring Systems (ITSs) focus on the students by offering them personalized feedback (VanLehn 2011; D'Mello et al. 2012; Heffernan and Heffernan 2014; Kulik and Fletcher 2016). ITSs need to be able to analyze student responses and, in most cases, compare them against a correct answer. The feedback given by such systems needs to be more meaningful than simply saying whether the response is correct or not. The drawback of most ITSs is that they still require a significant amount of human labor for every new question added to the system.

Recently, a new type of technology has emerged which has as its purpose increasing the interaction between teachers and students. Classroom Engagement Systems (CESs) allow all students to answer free response questions in classrooms, thus engaging all students at once. Comprehension SEEDING was introduced by Paiva et al. (2014) and has three primary goals: Self Explanation, Enhanced Discussion and INquiry Generation. These goals all work together to increase student engagement in classrooms. While CESs considers teacher and student feedback an important piece of the puzzle, the main focus is on the bigger picture of creating an environment where students are encouraged to participate in classroom discussions. Nevertheless, the system described by Paiva et al. (2014) adopts an over-simplified approach to the problem of formative assessment, using only lexical information to compare and organize responses before presenting them to the teacher.

Existing educational applications are often weak in at



Q: What protection measures do we take when we're doing experiments in the lab?

Reference Answer	Student Response
We wear gloves when handling substances and we put on goggles because eyes need protection.	Stuff could get in your eyes so you need to wear glasses at all times when dealing with substances.

MMPs: 1. We wear gloves when handling substances. → Not Understood

2. We put on goggles. → Understood

3. Eyes need protection. → Understood

Figure 1: Minimum Meaningful Proposition Entailment. The reference answer is broken down into its constituent MMPs. Entailment relations are established between each MMP and the student's response.

least one important aspect. Whether it is incomplete feedback to the teacher or to the student, or the need to develop question-dependent logic to assess students' responses, existing systems are not complete. This paper takes important steps towards filling in these gaps and developing educational applications that will be able to adapt to individual instructor needs, offer valuable and effective feedback to both teachers and students and, at the same time, scale and adapt to new questions without the need for expert intervention.

One way to significantly improve the next generation of educational applications is to enhance the feedback offered to teachers and students following a free-response question. This will increase the speed at which the instructors can assess the students' knowledge, make them aware of what concepts need to be covered more thoroughly before continuing with new material, and perhaps more importantly, facilitate real-time, organized and insightful feedback, leading to the generation of interesting classroom discussions that will engage more students. To accomplish this, simply determining entailment relations between a student's response and a reference answer as a whole is not sufficient. We need to encourage more deep questions that often require longer responses, and in such cases, the teacher's reference answer can easily discuss more than one concept. Student responses can entail an understanding of all the information in the reference answer or they can entail an understanding of only parts of it, leaving some concepts unaddressed. Thus, a finer-grained analysis of the reference answer, the student's response, and the response's relation to the reference answer is required.

On this account, we make use of Minimal Meaningful Propositions (MMPs) (Godea, Bulgarov, and Nielsen 2016). MMPs have recently been introduced as a decomposition of text into the set of propositions that individually represent single minimal claims or arguments that cannot be further decomposed without losing contextual meaning. By splitting the instructor's reference answer into MMPs, we can make more meaningful comparisons between the learner's answer and the individual claims expressed in the reference answer.

We take two different approaches to decide the entailment relations, one by using pre-trained word embeddings as input for a deep neural network, and the other by using hand-engineered features with an SVM classifier. The methods achieve similar results while seeing significant improvement in terms of F_1-score over a state-of-the-art system and two alternative approaches.

Figure 1 exemplifies our approach by showing a real classroom question, the teacher's reference answer and a student response. Our method first breaks down the reference answer into the three MMPs on the figure's left, and then predicts entailment relations between the learner's response and each of these reference answer MMPs. As can be seen, part of the student's response is correct, showing that s/he understood that the eyes need protection and goggles are a way to achieve this. However, s/he does not address the fact that wearing gloves is also an important protection measure while conducting experiments in the lab.

This example shows that instead of having to output a compromised entailment label for the whole reference answer, we can treat individual claims separately. If we combine this information with all the other students' responses similar to Godea, Bulgarov, and Nielsen (2016), the instructor can use it to generate insightful classroom discussions and correct most common misconceptions. Such an evaluation of the students' responses can significantly improve the feedback offered to teachers and students, reduce the time required to grade student responses and essays, and enhance the learning process in the classroom. Outside of educational applications, an approach like this will have applications in areas such as machine translation, complex query matching, summarization, information extraction, relation extraction and others (Dagan, Glickman, and Magnini 2006).

2 Related Work

Recognizing Textual Entailment (RTE) is the task of determining whether the meaning of a text (called *hypothesis*) can be inferred by another (Dagan, Glickman, and Magnini 2006). RTE is an important step in many Natural Language

Processing (NLP) applications where the diversity of natural language is of major importance, e.g., question answering, information extraction, etc. In the context of educational applications, RTE is the task of determining whether one sentence (or the student's response), entails another (or the instructor's reference answer). Most of the times, RTE in education differs because student responses in classrooms are often ungrammatical and use words or expressions that are not formal. RTE in classrooms is particularly important since many educational applications could benefit from more accurate algorithms, e.g., answer scoring, generating feedback, clustering student responses, etc.

Concept Rater (or c-rater) is an automated scoring engine developed by the Educational Testing Service (ETS) (Leacock and Chodorow 2003). The purpose of c-rater is to score short responses (up to 100 words) to open questions. In order to determine the paraphrase relation or the similarity, the system requires a set of reference answers as a model, input by an expert. There are four main steps in c-rater. First, in the Model Building phase, a set of correct answers are generated. This step is the most time consuming and labor intensive part of the process. The instructors have to enter separate sentences for each concept they use in their answer. Then, multiple paraphrases of the same sentence are being created manually, as well as synonyms of the concepts. To shorten the human effort of this approach the authors came up with a method that only requires manual concept-based scoring, generating the lexicon automatically (Sukkarieh and Stoyanchev 2009). The results indicate that the unweighted kappa values for the two approaches (manual and automatic) are "comparable" in 11 out of 12 scenarios, with the remaining scenario having the highest number of concepts, i.e., seven. Second, model answers and students' responses are processed using Natural Language Processing techniques and linguistic features are extracted. Third, using the features previously extracted, the Goldmap matching algorithm is used to automatically determine whether a student's response entails the model answer. Finally, scoring rules are applied to produce a score and feedback for the student. While c-rater needs human labor for every question added to the system, our proposed approach uses the same models, learned by machine learning, to split the reference answer into propositions and generate entailment labels for each proposition.

Nielsen et al. (2009) also thought about representing the reference answer as finer-grained constituents. They introduce the term *facets*, which are semantic components, roughly derived from typed dependencies. By breaking down the teacher's reference answer into facets, the authors can pinpoint the exact concept that the student understood, contradicted or didn't address. However, facets are often too fine grained to be meaningful on their own, out of the context of the proposition of which they are a part. Thus, entailing a semantic facet can be misleading with regard to the context of the question and the student's understanding.

SEMILAR is a tool introduced by Rus et al. (2013) which implements a number of word-to-word, sentence-to-sentence and document-to-document similarity measures. For our work, the most relevant are the sentence-to-sentence similarity measures which, among others, include: a semantic similarity scorer by Corley and Mihalcea (2005) and a Latent Semantic Analysis (LSA) (Landauer, Foltz, and Laham 1998) implementation using all of Wikipedia and the TASA corpus.

In Horbach et al. (2013), instead of focusing on the target answers supplied by instructors, the authors consider processing the text itself. This is the first use of reading texts for automatic short answer scoring in the context of foreign language learning. They show that, for German, simply using text-based features improves classification over models that only consider teacher authored responses.

SemEval-2014 included an entailment competition under the *Evaluation of Compositional Distributional Semantic Models on Full Sentences through Semantic Relatedness and Entailment* task (Marelli et al. 2014). Here, systems were presented with pairs of sentences and were evaluated on their ability to predict human judgments on semantic relatedness and entailment. The task attracted 21 teams, most of which participated in both subtasks. One team notes that by combining word overlap and antonyms one can detect 83.6% of neutral pairs and 82.6% of entailment pairs. The top-ranking systems in both tasks used compositional features and most of them also used external resources, especially WordNet. Almost all the participating systems outperformed the proposed baselines in both tasks.

Even though neural networks have been successful in detecting paraphrase relations (Hu et al. 2014; Yin and Schütze 2015), neural architectures often fail to obtain acceptable performance scores in RTE tasks due to the lack of large high-quality datasets. However, recently, Bowman et al. (2015) published the Stanford Natural Language Inference (SNLI) dataset, which due to its large size and high quality, allowed researchers to achieve high accuracy without handcrafted features. Using neural networks with long short-term memory units (LSTM), they reached an accuracy of 77.6% on this dataset. This was the first generic neural model without hand-crafted features to achieve performance close to that of a simple classifier using manually-engineered features for RTE. They accomplished this by encoding both sentences as fixed-length vectors and used their concatenation in a multi-layer perceptron for classification. Shortly after, Rocktaschel et al. (2015) altered Bowman et al.'s method by proposing an attentive neural network capable of reasoning over entailments of pairs of words and phrases by processing the hypothesis conditioned on the premise. By doing so, they achieved a higher accuracy on the SNLI dataset of 83.5%.

Chen et al., (2016) adopt an enhanced sequential inference model, which outperformed previous, more complicated network architectures. Their model uses bidirectional LSTMs (BiLSTM) for both local inference modeling and inference composition. Additional improvement is achieved by incorporating syntactic parsing information. This model sets the current highest performance on this dataset – 88.6%. The same performance is also reached by Wang, Hamza and Florian (2017), who used a bilateral multi-perspective matching (BiMPM) model. Given two sentences, their model first encodes them with a BiLSTM encoder and then

		Ref Ans MMPs		*Student Responses*			*Entailment Pairs*		
	Questions	*Total*	*Avg.*	*Avg.*	*Max.*	*Min.*	*Understood*	*Not Understood*	**Total**
Train	157	536	3.6	21.6	30	12	3380 (29%)	8305 (71%)	**11685**
Development	54	215	4	22.7	28	14	1418 (31%)	3204 (69%)	**4622**
Test	55	214	3.9	21.9	30	16	1573 (35%)	2935 (65%)	**4508**
Total	266	992	3.7	22.1	30	12	6371 (31%)	14444 (69%)	**20815**

Table 1: Dataset Statistics (the number of entailment pairs for a given question equals its number of reference answer MMPs multiplied by the number of student responses to the question)

matches them in both directions. Another BiLSTM layer is used to aggregate the matching results into a fixed-length vector. Based on it, a decision is made through a fully connected layer. Although reaching high accuracies on the SNLI dataset, these approaches are not appropriate for our data due to the small dataset size (20,000 vs 570,000 instances) and the nature of student responses (which are often ungrammatical).

3 Data

We use a modified version of the dataset introduced in Godea, Bulgarov and Nieslen (2016). The most important difference is adding the entailment labels. Statistics regarding the dataset can be found in Table 1. The questions in the dataset come from real middle school science classrooms, have an average of 22 student responses and come with a teacher supplied reference answer. The data was split into train, development and test sets, following the percentages: 60% train, 20% development and 20% test. The data was split at the question level (i.e., all of the instances of student responses for one question reside in a single split – the train, development or test dataset), which is why there are a different number of entailment instances for the development and test splits (i.e., the number of responses and MMPs varies). Two graduate students from the Education and Linguistics Department established the proper entailment relations between each pair of reference answer MMP and student response – *understood*, *misunderstood* or *not understood*, with a third annotator acting as an adjudicator. For the first two labels, annotators were required to mark associated evidence in the student response.

An instance in our dataset is derived from a pairing of a reference answer MMP and a student response. We treated the *misunderstood* instances as *not understood* due to the extremely low number of instances in this class (around 3%). There are a total of 20815 instances for this task, with about 30% of them being in the *understood* class and the remaining 70% being in the *not understood* class. In the future, we plan to expand our dataset and include all labels in the classification.

4 MMP Entailment

Rather than strictly checking whether the student's response is a paraphrase of, or entails the teacher's reference answer as a whole, we break the target conceptual knowledge into smaller propositions, or clauses (MMPs). This helps us separate complex structures in the reference answer and identify which specific propositions or clauses the student understood. In this section we perform experiments, show results and discuss the main errors occurred when predicting entailment relations between student responses and reference answer MMPs.

4.1 Classification

A first approach to this task is to use hand-crafted features. The features we used are described in Table 2 and are split into *General Features* and *Facet Features*. The 45 general features describe general relations between the reference answer MMP and the student response, such as the overall similarities and dependencies between words, Pointwise Mutual Information (PMI) scores, overlapping content, BLEU score (Papineni et al. 2002), etc. For the rest of the features, we make use of facets as described by Nielsen et al. (2009). Our decision to use facet information was motivated by their granularity level, allowing us to pinpoint the main relations between two texts. Features 46 through 183 encode information regarding one facet. They are computed and concatenated to the final feature vector 3 times: (1) for the least likely understood facet; (2) for the most likely understood facet; and (3), as the averages of all facets. The least and most likely understood facets are chosen based on the following algorithm: we automatically extract all facets from the reference answer MMP and compute the PMI similarity between its governors and modifiers and the governors and modifiers from all facets in the student's response. The facets with the lowest and highest average PMI score are chosen as the least and the most likely understood facets and are used in the feature extraction process.

Our second approach to this task is using word embeddings with a deep neural network. We used GloVe word embeddings with 50, 100 and 200 dimensions pre-trained on six billion tokens from Wikipedia 2014 and Gigaword 5 (Pennington, Socher, and Manning 2014). Specifically, we computed the average embedding vector for each text (reference answer MMP and student response), and combined them into a single vector by concatenating the element-by-element product vector and absolute difference vector (thus, experiments with 200-dimensional word embeddings resulted in a 400-dimensional input to the neural network).

General Features

1 - 10	Number of words, number of overlapping words and BLEU score for n = 1, 2, 3 and 4
11 - 13	Avg, max and min PMI score between reference answer (RA) MMP facets and student response facets
14 - 15	Student response contains animate/inanimate pronoun
16 - 35	Relatedness - overall similarity between words (stemmed and original) in the RA MMP and the student response (e.g.: fraction of exact matches, of zero co-occurrences and non zero co-occurrences, Maximum Likelihood Estimates (MLE) for the co-occurrence of the two terms)
36 - 45	Facet Similarities – overall similarity of facets in the RA MMP and the student response (similar to features 16- 35)

Facet Features

46 - 58	Governor detailed features (i.e., similarity features between the gov and its best match in the student response – POS features, MLE, best match features, etc.)
59 - 71	Modifier detailed features (i.e., similarity features between the mod and its best match in the student response – POS features, MLE, best match features, etc.)
72 - 73	Product between the similarity of the gov and mod and their best matching nodes in the student's response
74 - 79	Boolean features indicating whether the student response had any exact matches in the facet (gov/mod, stemmed/original)
80 - 94	Features of the path from the modifier to the governor and its best match, such as: direction, length, negations, comparisons between the paths, etc.
95 - 114	Features extracted from other facets that include the *modifier* in the current facet (similar to features 16-35)
115 - 134	Features extracted from other facets that include the *governor* in the current facet (similar to features 16-35)
135 - 144	Combined scores of all facets in which the gov or the mod occur (similar to features 16-35)
145 - 154	Combined scores of all facets on the path between the mod and the gov (similar to features 16-35)
155 - 157	Features of the best facet match in the student response
158 - 171	Pronoun coreference features
172 - 183	Relatedness scores of facets that express relations between higher-level propositions

Table 2: Feature Descriptions

4.2 Results

In Table 3 we report the precision, recall and F_1-score for each class (*understood* and *not understood*), as well as the weighted average F_1-score.

For comparison, we show the results obtained by a majority baseline, Latent Semantic Analysis (LSA), and Corley and Mihalcea's (2005) unsupervised system for measuring the semantic similarity of texts. The two latter approaches were computed using the SEMILAR toolkit (Rus et al. 2013). A score was obtained for each pairing of a reference answer MMP and a student response for the associated question. All pairs which obtained a score higher than a threshold t, were marked as understood. We used the development set to estimate the best value for t (LSA: t = 0.5; Corley and Mihalcea: t = 0.6). A state-of-the-art system, proposed by Horbach et al. (2013), was also tested for a more meaningful comparison. Even though their system was slightly altered to be applicable to our dataset, the main features remained unchanged:

- Lemma Overlap: two lemma overlap features. One normalized by the number of learner answer tokens, the other by the number of tokens in the MMP
- Dependency Triple Overlap: four features. Full match between dependency triples (modifier, dependency relation, governor) or a match between the two lemmatized words, both being normalized by either number of tokens (in the MMP or in the student answer)
- WordNet Similarity, using the aggregation methods proposed by Mohler and Mihalcea (2009) and Jiang and Conrath (1997)
- String Similarity: Levenshtein Distance
- Number of MMPs in the reference answer

Our reported pre-trained word embedding results were obtained using Keras (Chollet 2015) with TensorFlow (Abadi et al. 2015). All word embeddings experiments were enhanced with two additional features, computed by normalizing the number of identical lemmatized words by the number of words in the reference answer MMP and by the number of words in the student response, respectively. The best results on the development set were obtained using a feed forward neural network with two hidden layers, each having 64 hidden nodes. The dropout rate was set to 0.5, for both layers. Only a small number of iterations was needed to reach the results (between 10 and 20). A binary cross entropy loss function and a RMSprop optimizer were used to train the model. All parameters were tuned on the development set, while the reported results were obtained on the test set. A second approach uses the aforementioned manual features, which are fed to an SVM classifier.

As can be seen, the approach using word embeddings with 50 dimensions achieves the highest weighted average F_1-score of 0.76, performing about 13% better than the state-of-the-art system and the two alternative approaches. The SVM model using hand crafted features obtained a close F_1-score, of 0.73. However, we can observe important differences on the *understood* class where word embeddings models achieve a significantly higher F_1-score of 0.63. This is a notable improvement of about 43% over just using Latent Semantic Analysis, which only obtained an F_1-score of 0.44. In comparison with the state-of-the-art system, our ap-

	Understood			*Not Understood*			*Weighted Avg.*
$Model$	$Prec.$	$Recall$	F_1-score	$Prec.$	$Recall$	F_1-score	F_1-score
Majority Baseline	0	0	0	0.70	1	0.82	0.58
Latent Semantic Analysis	0.48	0.40	0.44	0.72	0.78	0.75	0.66
Corley and Mihalcea (2005)	0.50	0.37	0.43	0.72	0.81	0.76	0.66
Horbach et al. (2013)	0.61	0.29	0.39	0.75	**0.92**	**0.83**	0.67
SVM – manual features	**0.73**	0.41	0.53	0.76	**0.92**	**0.83**	0.73
Word Embeddings – 50 dim.	0.69	0.57	**0.63**	0.79	0.86	**0.83**	**0.76**
Word Embeddings – 100 dim.	0.63	0.58	0.60	0.78	0.82	0.80	0.73
Word Embeddings – 200 dim.	0.63	**0.64**	**0.63**	**0.81**	0.80	0.80	0.74

Table 3: MMP Entailment Results

proach is seeing an increase of 61% on the *understood* class. On the *not understood* class, the difference in results between the alternative approaches and our proposed methods is significantly lower, or none in the case of Horbach et al.'s approach. This is mainly due to the effectiveness of classifying instances in this class utilizing only the word overlap, which is generally very low for the *not understood* class.

4.3 Error Analysis

A challenge for our approach is that many of the questions in our dataset have multiple valid answers:

> *"Tell me what you know about acids, bases, salts, reactants, products and the neutralization process!"*

Questions like this can be answered correctly in more than one way without making any compromises. One possible approach would be for teachers to supply multiple correct responses, similar to the process followed in evaluating machine translation systems.

Our error analysis suggests that first classifying questions according to their expected answer type could substantially improve our ability to determine whether student understanding is entailed. Expecting a certain type of student responses, such as a short response (a noun phrase), free-response (a paragraph), an opinion, an enumeration, etc., could potentially increase our chances of success significantly. In such cases, a modified version of the entailment algorithm can be applied, and educational applications can treat questions differently, adjusting their feedback accordingly.

Another interesting observation made when looking at the data was regarding the word overlap between the reference answer MMPs and the student response. *Not understood* instances have a very low word overlap, making them easier to classify by straightforward baselines. On the other hand, word overlap for the *understood* instances varies greatly, reaching an average of only 1.8, when stemmed. While some understood pairs have a high overlap of over four identical content words, others may not have any content words in common. Furthermore, even when the word overlap is low, both *understood* and *not understood* pairs will contain related words, because generally, students will answer questions by talking about related concepts and ideas. This makes differentiating between the classes harder.

Moreover, whereas most RTE datasets contain formal texts, our student responses, and even some of the reference answers, are often ungrammatical or contain mistakes. For example, consider the following question (Q) and student response (SR):

> Q: *"What are fluids and how they affect motion?"*
>
> SR: *"i down kow, but snow it makes you go slower than you normally go, because snow is more conpact. fluids also effect motion because its harder talk through.fliuds are like watewr, snow, and rain"*

While some spelling mistakes can easily be fixed with an automatic spell checker, others depend on the context and they are harder to correct automatically: *"effect"* instead of *"affect"*, *"talk"* instead of *"walk"* and *"down"* instead of *"don't"*. This will change our word embeddings averages as well as the relations between the words and their sense as perceived by the classifiers. Such mistakes are often encountered as students are either unaware of the correct spelling or not paying enough attention:

> SR: *"yes because the giviel is weaker than oaw panteit"*
>
> SR: *"the elecricoll field around chaedthe magnet when a negitiv charge meets with a positive"*

To account for such errors we incorporated an automatic spell checker in our preprocessing phase. However, instead of simply correcting the wrong words, we first take into account all responses from the students, as well as words from the question and reference answer. Before correcting a misspelled word, we look to see which of its five most probable corrected forms were used by other students or the teacher. The stemmed version of words and the Levenshtein distance are also taken into account when comparing words. By doing so, we mitigate against replacing misspelled words with real words that have nothing to do with the context of the question.

		Understood			*Not Understood*			*Weighted Avg.*
$No.$	$Model$	$Prec.$	$Rec.$	F_1-score	$Prec.$	$Rec.$	F_1-score	F_1-score
1	SVM (WEs)	0.74	0.5	0.6	0.77	0.9	0.83	0.75
2	SVM (WEs + man. ftrs.)	073	0.54	0.61	0.78	0.89	0.83	0.76
3	DNN (man. ftrs.)	0.69	0.40	0.50	0.75	0.91	0.82	0.72
4	DNN (WEs + man. ftrs.)	0.71	0.45	0.55	0.75	0.90	0.82	0.73
5	SVM (man. ftrs. + LSA)	0.73	0.42	0.53	0.76	0.92	0.83	0.73
6	SVM (man. ftrs. + C&M)	0.73	0.42	0.53	0.76	0.92	0.83	0.73
7	SVM (man. ftrs. + LSA + C&M)	0.73	0.42	0.53	0.76	0.92	0.83	0.73
8	DNN (WEs + LSA)	0.61	0.32	0.42	0.71	0.89	0.79	0.66
9	DNN (WEs + C&M)	0.66	0.25	0.36	0.7	0.93	0.8	0.65
10	DNN (WEs + LSA + C&M)	0.6	0.32	0.42	0.71	0.88	0.79	0.66
11	**SVM (man. ftrs.)**	0.73	0.41	0.53	0.76	0.92	0.83	0.73
12	**DNN (WEs)**	0.69	0.57	0.63	0.79	0.86	0.83	0.76

Table 4: MMP Entailment Experimental Results (WEs - Word Embeddings 50 dim.; man. ftrs. = manual features; DNN = Deep Neural Networks; C&M = Corley & Mihalcea)

4.4 Further Experimentation

In our error analysis process, we also checked whether the learning algorithm or the features were the cause of the F_1-score increase. Moreover, since the manual features and word embeddings (WEs) are fairly independent of each other, combining them should, in theory, further improve the results. In addition, we also experimented with adding LSA and Corley & Mihalcea's scores to our existing feature sets. These results are shown in Table 4. As can be seen comparing rows 1 and 11, the SVM achieves a substantially higher F_1-score on the *understood* class using WEs instead of the manual features. In fact, adding WEs to our best SVM approach (row 2) results in a weighted average F_1-score of 0.76, which is equal to that of the deep neural network (DNN), (row 12). In contrast, adding the handcrafted features to the DNN (row 4), substantially decreases the results on the *understood* class. Experiments were also performed where the weights for WEs and manual features were separately learned by individual DNNs and merged at a later stage into a third DNN. These results did not exceed those obtained in row 12. A conclusion that can be drawn from these experiments, consistent with what we initially saw in Table 3, is that WEs are more helpful than manual features, particularly in identifying the minority *understood* class. Moreover, even though they seem independent of each other, combining manual features and WEs does not offer much improvement.

Rows 5 through 10 show experiments when either LSA, Corley & Mihalcea (C&M), or both scores are added to our best approaches (rows 11 and 12). For SVM, adding these measures offers no change in the final F_1-scores. On the other hand, the results drop significantly when adding these features to DNNs. This might happen because the DNN puts too much weight on these scores, which are more informative than individual WE dimensions, and hence, fails to properly learn from the WE features.

5 Conclusions and Future Work

We believe that accurately analyzing the students' responses and efficiently comparing them against the instructor's reference answer is the key to effective real-time and domain-independent educational applications. Moreover, a thorough, fine-grained comparison of the two answers can open up a variety of new possibilities to enhance feedback for both teachers and students.

To this end, this paper makes use of Minimal Meaningful Propositions in order to break down complex structures and to perform a fine-grained analysis of student responses. This is the first work to do so in a fully automatic process. We presented two different methods for detecting the entailment relation between a student response and a reference answer MMP: one that used features focused on semantic facets and an SVM classifier and the other which used word embeddings input to a deep neural network. These approaches exceeded the performance of the most frequent class baseline by 31% and a state-of-the-art system and two alternatives by approximately 15%, achieving a weighted average F_1-score of 0.76.

In the future, we plan to increase the dataset size so that our classifiers can learn to predict the severely underrepresented *misunderstood* class. We also plan to develop question and reference answer classifiers in order to distinguish between the types of questions and responses expected by the instructors. In the longer term, these models will be integrated into an educational applications in order to improve the learning process through more meaningful targeted feedback.

6 Acknowledgements

This research was supported by the Institute of Education Sciences, U.S. Department of Education, Grant R305A120808 to University of North Texas. Opinions expressed are those of the authors.

References

Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mané, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viégas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; and Zheng, X. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Bowman, S. R.; Angeli, G.; Potts, C.; and Manning, C. D. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; and Jiang, H. 2016. Enhancing and combining sequential and tree lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.

Chollet, F. 2015. Keras. https://github.com/fchollet/keras.

Corley, C., and Mihalcea, R. 2005. Measuring the semantic similarity of texts. In *Proceedings of the ACL workshop on empirical modeling of semantic equivalence and entailment*, 13–18. Association for Computational Linguistics.

Dagan, I.; Glickman, O.; and Magnini, B. 2006. The pascal recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*. Springer. 177–190.

D'Mello, S.; Olney, A.; Williams, C.; and Hays, P. 2012. Gaze tutor: A gaze-reactive intelligent tutoring system. *International Journal of human-computer studies* 70(5).

Godea, A.; Bulgarov, F.; and Nielsen, R. 2016. Automatic generation and classification of minimal meaningful propositions in educational systems. In *Coling 2016*.

Heffernan, N. T., and Heffernan, C. L. 2014. The assistments ecosystem: building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education* 24(4):470–497.

Horbach, A.; Palmer, A.; and Pinkal, M. 2013. Using the text to evaluate short answers for reading comprehension exercises. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 1, 286–295.

Hu, B.; Lu, Z.; Li, H.; and Chen, Q. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, 2042–2050.

Jiang, J. J., and Conrath, D. W. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*.

Kulik, J. A., and Fletcher, J. 2016. Effectiveness of intelligent tutoring systems: a meta-analytic review. *Review of Educational Research* 86(1):42–78.

Landauer, T. K.; Foltz, P. W.; and Laham, D. 1998. An introduction to latent semantic analysis. *Discourse processes* 25(2-3):259–284.

Leacock, C., and Chodorow, M. 2003. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities* 37(4):389–405.

Marelli, M.; Bentivogli, L.; Baroni, M.; Bernardi, R.; Menini, S.; and Zamparelli, R. 2014. Semeval-2014 task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. *SemEval-2014*.

Mitchell, T.; Russell, T.; Broomhead, P.; and Aldridge, N. 2002. Towards robust computerised marking of free-text responses.

Mohler, M., and Mihalcea, R. 2009. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, 567–575.

Nielsen, R. d.; Ward, W.; and Martin, J. h. 2009. Recognizing entailment in intelligent tutoring systems*. *Nat. Lang. Eng.* 15(4):479–501.

Paiva, F.; Glenn, J.; Mazidi, K.; Talbot, R.; Wylie, R.; Chi, M. T.; Dutilly, E.; Helding, B.; Lin, M.; Trickett, S.; et al. 2014. Comprehension seeding: Comprehension through self explanation, enhanced discussion, and inquiry generation. In *Intelligent Tutoring Systems*, 283–293. Springer.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311–318.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, 1532–1543.

Rocktäschel, T.; Grefenstette, E.; Hermann, K. M.; Kočiskỳ, T.; and Blunsom, P. 2015. Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

Rus, V.; Lintean, M. C.; Banjade, R.; Niraula, N. B.; and Stefanescu, D. 2013. Semilar: The semantic similarity toolkit. In *ACL (Conference System Demonstrations)*, 163–168. Citeseer.

Sukkarieh, J. Z., and Stoyanchev, S. 2009. Automating model building in c-rater. In *Proceedings of the 2009 Workshop on Applied Textual Inference*, 61–69. Association for Computational Linguistics.

VanLehn, K. 2011. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist* 46(4):197–221.

Wang, Z.; Hamza, W.; and Florian, R. 2017. Bilateral multi-perspective matching for natural language sentences. *arXiv preprint arXiv:1702.03814*.

Yin, W., and Schütze, H. 2015. Convolutional neural network for paraphrase identification. In *HLT-NAACL*, 901–911.