Automatic Generation and Classification of Minimal Meaningful Propositions in Educational Systems

Andreea Godea, Florin Bulgarov and Rodney Nielsen
Department of Computer Science and Engineering
University of North Texas, TX, USA
{AndreeaGodea, FlorinBulgarov}@my.unt.edu
Rodney.Nielsen@colorado.edu

Abstract

Truly effective and practical educational systems will only be achievable when they have the ability to fully recognize deep relationships between a learner's interpretation of a subject and the desired conceptual understanding. In this paper, we take important steps in this direction by introducing a new representation of sentences – Minimal Meaningful Propositions (MMPs), which will allow us to significantly improve the mapping between a learner's answer and the ideal response. Using this technique, we make significant progress towards highly scalable and domain independent educational systems, that will be able to operate without human intervention. Even though this is a new task, we show very good results both for the extraction of MMPs and for classification with respect to their importance.

1 Introduction

Over the last few decades, technology has provided us many powerful tools that have completely changed our daily routines. However, one crucial area where technology has yet to have the significant impact suggested by its true promise is in education. Most students around the world have been learning in the same manner for decades. Nevertheless, in the past few years technology has started to increase its role in the learning process and has begun improving the effectiveness of students and instructors. Several groups are developing tools or systems with the goal of improving the feedback provided to students and instructors, assessing students' understanding of a concept, and facilitating their self-guided learning.

Intelligent Tutoring Systems (ITSs) were created with the goal of improving learning through real-time and personalized feedback for students (Graesser et al., 2001; Rosé et al., 2003; Makatchev et al., 2004; Pon-Barry et al., 2004). ITSs need to be able to interpret complex student responses and improve their feedback as they process more questions and responses, but essentially all existing systems require skilled developers to write new rules or train new classifiers for each additional question. Generally, an ITS only provides feedback to students, and when they do provide feedback to instructors, it is typically just high-level information regarding the correctness of the answer. Much of the prior work in this area originated in educational assessment systems (Mitchell et al., 2002; Sukkarieh et al., 2003; Nielsen et al., 2009). Most such systems investigate similarity or entailment relationships between a learner's answer and the reference answer, and then communicate a score to the teacher.

More recently, a new type of educational technology has emerged with the goal of increasing student engagement in classrooms (Paiva et al., 2014). Classroom Engagement Systems (CES) are meant to replace audience response (clicker) systems (Duncan, 2006; Fies and Marshall, 2006) by allowing students to construct answers to free-response questions. Unlike an ITS, the main goal of a CES is to facilitate teacher-student interaction. However, the system Paiva and associates present has some notable weaknesses: the analysis is strictly lexical, all content words are treated with equal importance, and only a small number of student responses are chosen as representatives.

The lack of tools to precisely identify the importance of concepts in the reference answer without manual intervention for each question, and the lack of tools to analyze the nature of a student's response,

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 3226–3236, Osaka, Japan, December 11-17 2016.

Q: Explain how mass is different than weight.

RA: Mass tells you how much matter an object has. Weight tells you how gravity affects the mass and it changes when gravity changes.

Reference Answer MMPs:		**Student Answer** MMPs:
1. Mass tells you how much matter an object has.^{P}	$\Longleftarrow$	1. Mass is the amount of matter an object holds.
2. Weight tells you how gravity affects the mass.^{P}		2. Weight is how heavy/light something is.
3. Weight changes when gravity changes.^{S}	$\Longleftarrow$	3. Weight gets heavier in higher gravity.

Figure 1: MMP Overview. **P** refers to a primary MMP while **S** denotes a secondary MMP. The entailment symbol signifies that the student understood that MMP.

again without manual coding per question, are significant weaknesses in existing educational technology. To that end, this paper takes important steps to address those weaknesses, introducing methods that will enable educational systems to effectively analyze deep semantic relationships between a learner's answer and a reference answer. Our primary contributions are:

- We introduce the concept of Minimal Meaningful Propositions (MMP), a decomposition of text, such as a question's answer, into the set of propositions that individually represent single minimal claims or arguments that cannot be further decomposed without losing contextual meaning, and taken as a whole represent the entire meaning of the text.
- We present a computational method for breaking text down into its MMPs.
- We present a method, features and categories to classify a reference answer's MMPs, which will allow educational systems to ensure feedback is focused on the most pertinent points.

MMPs are extracted from the reference answer to enable a thorough comparison between it and a student's response in order to diagnose which concepts were understood, misunderstood, or omitted from the response, as well as to determine the importance of those concepts. The research described in this paper will represent a strong foundation for the next generation of fully automated scoring systems. A complete example showing how MMPs are useful is given in Figure 1. Here we show a question, the reference answer, its MMPs, the student answer MMPs and their entailment relations. As can be seen, the student fails to address primary MMP 2 from the reference answer. However, the student successfully understood the concepts expressed in MMPs 1 and 3.

The final outcome of this approach to analyzing student responses will open a variety of new possibilities for fully automated educational systems. For instance, it will support: improved dynamic analysis of student answers to novel questions, the ability to focus on the most important conceptual misunderstandings, the means to provide meaningful feedback to instructors regarding the classroom understanding of concepts, and a construct for more effectively grouping similar answers either for realtime classroom analysis or for assessment purposes. This will allow such systems to be flexible enough to adapt to individual student and teacher needs and to various pedagogical methods. MMPs could also be an effective level of analysis in a wide variety of other NLP applications such as summarization, translation and more general textual entailment. In the following sections we describe the MMP concept and present our methods and results for MMP Extraction and MMP Classification.

2 Related Work

The goal of our work is not only to research means to better assess students' answers in a classroom environment, but also to research tools for more effective and constructive feedback regarding overall understanding of a subject. Although we are the first to introduce the concept of Minimal Meaningful Proposition, other works in the literature have had relatively similar goals (Burrows et al., 2015).

C-rater, a scoring engine developed by ETS, grades a student's answer to assessment questions (Leacock and Chodorow, 2003). C-rater recognizes paraphrases of a set of reference answers to determine wether the student's answer is correct. Although much of the work done by c-rater has been automated in the past years (Sukkarieh and Stoyanchev, 2009), it still requires an appropriate set of responses that have

Data	#Questions	Answ. Words/Question	#MMP	MMP/Q	Primary	Secondary	Extraneous	Redundant
Train	208	25.5	826	4.0	676 (81.8%)	100 (12.1%)	41 (4.9%)	9 (1.0%)
Test	109	22.1	383	3.5	323 (84.3%)	45 (11.7%)	12 (3.1%)	3 (0.7%)
Total	317	24.3	1209	3.8	999 (82.6%)	145 (11.9%)	53 (4.3%)	12 (0.9%)

Table 1: MMP counts and the average length of the reference answers in words.

already been holistically scored by trained raters.[1] In contrast, our approach is fully automated and can be used in a dynamic setting to recognize the focused relationships between a specific reference answer proposition and the student's response. MMPs are also classified for importance in a fully automated, domain-independent fashion using general linguistic features extracted from the reference answer, the question, and their interrelationships.

Another approach with a similar goal is entailment of semantic facets (Nielsen et al., 2009). Here, rather than checking whether the student's answer is a paraphrase of the reference answer as a whole, the authors break the target conceptual knowledge down into fine-grained *facets*, derived roughly from the typed dependencies in a parse of the reference answer. This might allow pinpointing the facet of the reference answer that the student misunderstood at a very fine-grained level, but unlike Minimal Meaningful Propositions, facets are often not *meaningful* without much more context. Hence, entailment of a semantic facet could be misleading with regard to student understanding.

Other related concepts have been introduced in text summarization, question answering and dialog generation. For example, *Elementary Discourse Units* (EDUs), developed for discourse segmentation, are defined as minimal non-overlapping textual spans representing units of discourse. EDUs are generally used as a precursor to identifying relationships between discourse segments. Previous work extracting EDUs has proposed rule-based approaches (Polanyi et al., 2004), classification of discourse boundaries (Soricut and Marcu, 2003; Subba and Di Eugenio, 2007; Afantenos et al., 2010) and sequence labeling (Hernault et al., 2010). However, in contrast with MMPs, EDUs are not necessarily either minimal or meaningful – for example, conditionals required for *meaningful* interpretation of a proposition are not included in the same EDU as their consequent, and a "minimal" discourse unit text span can often be broken into multiple finer-grained *minimal* propositions.

Nenkove and Passonneau (2004) introduced the *Summary Content Unit* (SCU) as a key component of the Pyramid evaluation method for multi-document summarization. SCUs are defined as semantically-motivated, sub-sentential units of variable length and emerge from the annotation of multiple human summaries for the same input. Previous work extracts SCUs manually (Nenkova and Passonneau, 2004; Nenkova et al., 2007) or uses topic modeling to match topics with manually-extracted SCUs (Hennig et al., 2010). However, to the best of our knowledge, there is no model for automatically constructing SCUs. Another important difference is the input data being used. SCUs are extracted from multiple, well-structured human summaries; whereas, MMPs emerge from a single version of a potentially poorly-structured answer. A review of SCUs also finds that, while they are *meaningful*, they are not necessarily *minimal* – many SCUs are syntactically complex and would be divided into multiple MMPs.

3 Data Description

The data used in our experiments consists of 317 questions that were asked in real science classes from middle school. Each question comes with the teacher's reference answer, which was decomposed into MMPs by two graduate students (from education and science major) and adjudicated by a third (from Education and Linguistics). Each of the first two annotators labeled the data independently and the adjudicator decided the correct label among the existing annotations.

The first stage of the annotation was identifying the MMPs in the instructors' reference answers. Annotators were provided guidelines for restating a reference answer as its corresponding set of minimal meaningful propositions, or distinct stand-alone claims, and given several guiding examples.

The second stage of the annotation process was to classify the MMPs into one of the following classes:

[1] `https://www.ets.org/research/topics/as_nlp/written_content/`

1. *primary*: fundamental to answering question
2. *secondary*: relevant but not integral to answer – often clarify or qualify a primary MMP
3. *extraneous*: unnecessary or minimally relevant to the question
4. *redundant*: contain information directly or indirectly provided by the question

As can be seen in Table 1, 1209 MMPs were annotated in total, with an average of 3.8 MMPs per answer. 999 MMPs were annotated as primary, 145 as secondary, 53 as extraneous and 12 as redundant. Training and test sets were created by randomly splitting the questions (2/3 in training and 1/3 in test).

4 Minimal Meaningful Propositions

Consider a sentence to be comprised of a set of related propositions. We define an MMP as a *proposition* that cannot be broken down into finer-grained propositions (it is *minimal*) and still be interpretable without further context (it is *meaningful* on its own). A sentence usually contains more than one MMP. Note that the MMPs only state explicit propositions, not any implications, presuppositions, or entailments. Moreover, in our case, an MMP should relate to the question in a way or another, when treated independently. The goal of the present work is to automatically extract MMPs from a question's reference answer and classify them according to their importance. The example below is a real question and reference answer asked in a classroom, and its human-extracted MMPs.

Q: How did Rutherford figure out that atoms are mostly empty space, and that the nucleus is positive?

RA: He used gold foil hammered about an atom thick, and placed radium in a lead lined box that emitted positive alpha particles towards the gold foil.

MMPs:

1. Rutherford used gold foil with the thickness of an atom.
2. Rutherford placed radium in a lead lined box.
3. The lead lined box emitted positive alpha particles towards the gold foil.

Extracted MMPs can contain information that was initially spread out over the sentence. These finer-grained propositions allow us to separate the different pieces of information expected from a student and classify their importance. An educational system that successfully uses MMPs will be able to tell the teacher which concepts were understood, contradicted, or omitted by the students.

5 MMP Extraction

A high level summary of the MMP extraction process is as follows. First, in the learning phase, we learn the unique set of syntactic patterns covering all of the gold-standard human generated MMPs. Then, in the application or testing phase, we process sentences recursively, on each call extracting the MMP associated with the longest matching pattern learned from the training set and recursively processing the remainder of the sentence. If part of the sentence remains and no further patterns match, the remainder forms the final MMP. The details of this process follow.

In the *learning* phase, the algorithm learns structural templates from a shallow parse (i.e., chunks) of the gold-standard MMPs in the training data. These templates will be used to extract MMPs from test set answers. Figure 2 shows an MMP and the structure extracted. Table 2 shows the most frequent structures in the dataset. The frequencies follow a Zipfian distribution, with the five most common structures covering almost 50% of the MMPs.

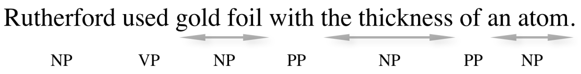

Figure 2: MMP Structure

Rank	Structure	%
1	*NP, VP, NP*	19.5
2	*NP, VP, NP, PP, NP*	11.3
3	*NP, VP, PP, NP*	8.5
4	*NP, VP*	7.6
5	*NP, PP, NP, VP, NP*	2.6

Table 2: Most Frequent MMP Structures

In the *test* phase, the algorithm first splits the answer into sentences, which it parses using Stanford CoreNLP (Manning et al., 2014). Then conjunctions are automatically pre-processed to: replace enu-

	F_1-score			BLEU score			
	Avg. P	Avg. R	F_1	1-grams	2-grams	3-grams	4-grams
MMP Extraction	**0.725**	**0.552**	**0.627**	**0.569**	**0.495**	**0.360**	**0.172**
Predicates Baseline	0.437	0.375	0.404	0.290	0.229	0.160	0.107
Sentence Baseline	0.438	0.449	0.443	0.377	0.315	0.246	0.154

Table 3: MMP Extraction Results

merations with a single base phrase type, and split conjoined SVO (i.e., Subject Verb Object) structures into separate sentences, replicating the subject and verb as appropriate.

Then, for each sentence, the algorithm finds the longest structure matching a pattern learned during training. If the matching pattern only covers a portion of the sentence, the algorithm extracts that portion as an MMP and recursively processes the unmatched portion of the sentence. If any base phrases remain unmatched when the recursion bottoms out, they become the final MMP. Due to the nature of the algorithm, our method is generalizable to different questions types or domains.

Given our example question **Q** in Section 4, the automatically extracted MMPs are:

1. He used gold foil hammered about an atom thick.
2. Placed radium in a lead lined box.
3. Emitted positive alpha particles towards the gold foil.

As can be seen, the automatically extracted MMPs are very similar to the human-generated examples. The major difference being the missing subject in the last 2 MMPs, which we will address in future work.

5.1 Results

To evaluate the performance, we compared the set of system-generated MMPs with the gold standard for each question. We pre-processed both system-generated and gold-standard MMPs to remove stop words and stemmed the remaining words using the Porter Stemmer. We report the Precision, Recall and F_1-score as well as the BLEU score. Precision is computed as the number of matching words divided by the total number of words in the system-generated MMP. Recall is the number of matching words divided by the total number of words in the gold-standard MMP. The BLEU score, introduced in (Papineni et al., 2002), is a highly-adopted method for automatic evaluation of machine translation systems. BLEU is based on a modified computation of *precision*, using the number of matching n-grams. It ranges from 0 to 1, with values closer to 1 representing more similar texts. Using different values of n, we can measure different aspects of the evaluation, *adequacy*: n = 1-2, and *fluency*: n = 3-4.

MMP-level metric values are based on a greedy iterative alignment of system-generated and gold-standard MMPs, where on a given iteration the algorithm aligns, processes, and then removes the pair with the highest F_1-score (or BLEU score). MMP-level values are averaged to get a question-level value, and finally, Table 3 presents the average over all questions.

For comparison we also computed two baselines. The *Sentence Baseline* is a method in which every sentence of the reference answer is treated as an MMP. In the *Predicates Baseline*, we build an MMP for every predicate in a sentence. Using SENNA (Collobert et al., 2011), we then identify the predicate's arguments and attach them to the MMP. As can be seen, the method that we propose, *MMP Extraction*, significantly outperforms the two baselines, achieving an F_1-score of 0.627 and showing that the pattern matching approach generalizes well on new, unseen questions. The precision is higher than the recall, meaning that the system generated MMPs are shorter than the human-generated ones.

The BLEU score achieved for different n-gram sizes are also considerably higher than the baselines'. The *adequacy* scores, unigrams and bigrams, are fairly high for this new task. As we increase the number of consecutive words to be scored, the score drops. This is a normal behavior for tasks where different solutions to a problem exist without any compromises. When using the BLEU metric to score the *fluency*, trigrams and 4-grams, it is strongly recommended that you have more than one reference solution or, in our case, human-annotated MMPs. When a single reference solution exists, as in our case, substantially lower values are expected (Papineni et al., 2002). For comparison, in a translation task, on a test corpus of about 500 sentences, a human translator scored 0.346 against four references and scored 0.257 against two references, when $n = 4$ (Papineni et al., 2002).

5.2 Error Analysis

Some of the most common errors occurring in the extraction phase are associated with the subjectivity of the task. Consider the following reference answer and its system-generated MMPs:

RA: *Once light reaches our eyes, signals are sent to our brain and our brain deciphers the information in order to detect the appearance, location and movement of objects.*

1. Light reaches our eyes.
2. Signals are sent to our brain.
3. Brain deciphers the information.
4. In order to detect the appearance, location and movement of objects.

First, in the human-annotated set of MMPs, 1 and 2 are joined into a single proposition: *When light reaches our eyes, signals are sent to our brain.* The annotators believed that the two pieces of information were too dependent to be separated. On the other hand, the system found two different claims and therefore, extracted two propositions. In addition, the fourth MMP generated by the system is dependent on the context and thus, not very good. The annotators broke this piece of text into three different MMPs, one for each element detected by the brain. In future, we plan to solve this issue using the semantic roles of constituents. We will also include coreference resolution to improve detecting agents.

Other errors made by the system can be fixed by including more syntactic and semantic information. For example, in a student response where the object does not immediately follow the predicate, our algorithm can get confused, and in some cases will not include the object as part of the MMP. In future work, we will check the validity of verb usages in an external resource such as VerbNet (Schuler, 2005), which will help us distinguish, for example, between transitive and intransitive verbs.

6 MMP Classification

Given a question, reference answer and its MMPs, our goal is to identify the importance of each MMP with respect to the question. We follow a supervised approach to classify MMPs as *primary, secondary, redundant* or *extraneous*. Many of the features the classifier uses to determine this importance compare the MMPs to the question. Hence, in preparation for feature extraction, we pre-process the question to eliminate unnecessary information and identify its key concepts. We then extract a number of features and train a classifier to predict labels for each MMP.

6.1 Question Pre-processing

Questions often include instructions to guide students on issues unrelated to content (e.g. *answer in* $\leq$ *2 sentences*). An analysis of such text revealed it can provide unnecessary errors to our model. Therefore, we filtered out text matching patterns indicative of such instructions. Three cases were explored:

Hints. We filtered out sentences starting with the keyword *hint*, if the hint was unrelated to other question content (e.g. Why do models change over time? *Hint: think about why your idea changed.*). The hint sentence was deemed related if the Pointwise Mutual Information (PMI) between any of its words and those of the rest of the question exceeded a threshold of $T = 0.28$, which was learned from annotated word pairs (*related* vs. *not related*) from the training data. In the example above, the hint was removed, since no relationship was found with the preceding sentence.

Punctuated Instructions. While analyzing the training data, several additional patterns indicating content-independent instructions were discovered. For example, in the question: *"Using the tables, describe how you know the object is accelerating."*, the instruction *using the tables* is unrelated to understanding the core *object acceleration* concept. In a substantial number of patterns, the instructions were delimited by punctuation, as in this case. Rules were developed to detect and filter out these punctuation-delimited content-independent instructions.

Parentheticals. Parenthesized text often includes important abbreviations, definitions or examples, but it can also contain content-independent instructions. Based on analysis of the training data, we wrote rules to filter out parentheticals that: 1) contain imperative statements: *. . . (Do not touch your eyes!)*, 2) start with negation: *What gives an atom its VOLUME (not its mass)?*, or 3) do not contain a noun, since this is highly correlated with instructions: *... the dependent variable (what we measure).*

6.2 Key Concept Identification

We identify a subset of the key concepts in the question for use in feature extraction. Specifically, we focus on concepts identifiable using part-of-speech tags and grammatical dependency relations, which covers the vast majority of key concepts. Generally key concepts are expressed as nouns, but many are also expressed as adjectives (*Explain what* homogeneous *means*) or gerunds (*Explain what* weathering *means*). The most common dependency types associated with key concepts are: adjectival modifiers (`amod`), noun compound modifiers (`nn`), and copulas (`cop`). However, we only consider dependency word pairs that are collocations as determined by applying the *Likelihood Ratio* statistic, using word counts from a large corpus consisting primarily of *Gigaword* (Graff and Cieri, 2003). If a dependency pair is not a collocation only constituent words matching previous selection criteria are used (e.g., since the dependency concept *equal forces* is not a collocation, only *forces* is retained). Given the question: *What evidence can indicate if a change is physical?*, the system identifies the key concepts: *evidence* and *physical change*.

6.3 Classification

To classify MMPs, we follow a supervised approach, training Random Forests on features indicative of the classes. However, since the original data lacks of *redundant* MMPs (see Table 1), we enriched the *training* data by adding 64 manually generated examples. Following the patterns in the training data, we created redundant MMPs by using information already stated in the question or paraphrasing parts of it.

Feature Engineering. Using information from the training data, we manually designed features based on the question, the MMPs, the reference answer, and the relations between them. From all the features explored, we chose the set that performed best in 10-fold cross-validation (10xCV) on training data. The final set of features is described in Table 4.

Note that two versions of the question were considered in this process: the original question (Q) and a version based strictly on its interrogative and imperative sentences (Q′). Various types of features were explored: 1) *features derived from the question representation* – $\{1, 2, 4\}$, 2) *semantic features* – $\{5, 18, 25, 27\}$, 3) *syntactic features* – $\{6, 7, 8, 9\}$, 4) *features focused on mismatches between the MMP and question* – $\{20, 21, 26\}$ and 5) *features focused on overlapping information* – $\{3, 10, 13, 14, 15\}$.

QUESTION FEATURES	
1.	Q contains structures similar to *"tell me what you know"*.
2.	Q has only one concept.
3.	Q has only one concept and it appears in MMP.
4.	Q has only one MMP attached.
5.	Q′'s concepts have hyponyms/hypernyms in MMP. (Fellbaum, 1998)
6.	Q′'s subject is lemma in MMP.
7.	Q′'s subject is concept in MMP.
8.	Q′'s subject is subject in MMP.
9.	Q′'s predicate appears in MMP.
MMP FEATURES	
10.	#overlapping lemmas between MMP and Q, excluding stop words.
11.	max PMI of MMP words (not in Q) and Q words.
12.	min PMI of MMP words (not in Q) and Q words.
13.	all MMP words appear in Q.
14.	all MMP words appear in a Q's sentence.
15.	all MMP words appear in a Q's interrogation.
16.	at least one MMP concept addressed by Q.
17.	MMP's predicate occurs in Q.
18.	MMP's predicate has hyponyms/hypernyms in Q.
19.	MMP's subject is subject in an answer sentenceaddressing Q.
20.	MMP provides a reason, although Q did not ask for it.
21.	MMP provides more information than required by Q.
22.	MMP is relevant for Q.
23.	the current MMP embeds other MMP.
24.	MMP's subject in Q′'s concepts.
25.	MMP's subject relates to Q′'s concepts using hyponymy or PMI.
26.	MMP's subject not in Q′, but refers to previous MMPs.
27.	max similarity score between MMP and Q′'s sentences.

Table 4: Description of Features

6.4 Results and Discussion

We report *P*recision, *R*ecall and F_1-score per class, using both a *strict* evaluation, based on adjudicated labels, and a *relaxed* evaluation, where the system is credited for matching either annotator's label. The results from 10xCV on the training data are presented in Table 5. As can be seen, *primary* and *redundant* are the best performing classes – they follow more recognizable patterns than the others. Whereas, *secondary* is the worst performing class. The vast majority of system errors on the *secondary* class are predictions of *primary*. Similarly, in analyzing the human annotation disagreements for MMPs adjudicated as *secondary*, the confusion was with *primary* in the vast majority of the cases – discriminating between *secondary* and *primary* is hard for humans as well as the system. This suggests there is more of a continuum between *primary* and *secondary* rather than a sharp decision boundary. This will be explored further in future work.

While the increase in the F_1-score for the *relaxed* evaluation is more than 6% for *secondary*, when viewed as a relative reduction in the error rate, the increase is considerable for all classes, ranging from 10% to 27%. This provides further motivation for a future investigation into the similarity in the errors in system and human judgements. Despite challenges, the high system F_1-score demonstrates the feasibility of the task and the promise of the system.

Next, we compare the performance to the majority class baseline, where each instance is classified as *primary*. Table 6 reports the weighted F_1-score achieved on both the test set and in 10xCV on training. The proposed approach outperforms the baseline in both scenarios. Our method shows an improvement of almost 16% in 10xCV compared with the baseline. Employing the *relaxed* evaluation, our performance is higher than the baseline with almost 16% on 10xCV and 7% on the *test* set.

The system outperformed the baseline under all scenarios (Table 6). It shows a slight improvement on test, even though the baseline results are 12% higher on test than on training. This difference in the baseline is due to the addition of redundant MMPs to the training data, as detailed earlier.

	Strict Matching			Relaxed Matching		
Classes	P	R	F_1	P	R	F_1
Primary	0.840	0.970	0.900	0.884	0.976	0.927
Secondary	0.675	0.250	0.365	0.835	0.303	0.429
Redundant	0.875	0.662	0.753	0.856	0.797	0.810
Extraneous	0.882	0.365	0.517	0.950	0.514	0.588

Table 5: Training Set 10xCV Results

	Training 10xCV		Test	
System/Approach	Strct F_1	Rlxd F_1	Strct F_1	Rlxd F_1
Proposed Approach	**0.810**	**0.857**	**0.815**	**0.876**
Majority Class	0.653	0.701	0.770	0.804

Table 6: System weighted F_1-score vs. Baseline

Ablation study. To assess the contribution of different categories of features, we performed an ablation study. Table 7 reports the weighted F_1-score when training on *all* of our features, and when training on all features except the specified set. The final column shows the increase in the error after removing the feature set, as a percent of 0.185 – the error when training on all features.

All feature categories have a substantial contribution to the results. The features derived from *Mismatching Information* between the question and MMP are especially useful – their removal results in a 21% relative increase in the error. The *secondary* and *extraneous* classes suffer the largest increase in error when *Mismatching Information* is not leveraged. This is logical since the *extraneous* information is not directly related to the question and *secondary* MMPs are optional and less directly tied to the question than *primary* MMPs. *Semantic features* have the second largest impact. While they help all classes, the greatest impact is on *primary* and *secondary* classes. MMPs with these classes contain highly relevant information for the question and the semantic features help identify indirect semantic relationships between parts of the MMP and the question. *Syntactic features* also have a positive contribution, especially for the *redundant* class. This is likely due to a more regular pattern in the mapping between components of a redundant MMP and the syntactic structure of the question. The *Question Representation features* also make a substantial contribution, particularly in classifying *primary*, *secondary* and *extraneous* MMPs. For example, if the question contained a single key concept and the MMP did not address it, the MMP is probably *extraneous*; whereas, if the MMP is strongly related to the key concept, then it is likely *primary* or *secondary*.

Feature Category *Removed*	**Wtd F_1**	**RIE%**
None (trained on *all* features)	0.815	–
Syntactic Features	0.800	8
Question Features (Q)	0.794	11
Question Representation Features	0.790	14
Semantic Features	0.787	15
Mismatching Information	0.776	21

Table 7: Feature ablation: weighted F_1 with *relative* percent increase in error (RIE) on Test

7 Conclusion

This work has resulted in three notable contributions. First, we introduced the concept of Minimal Meaningful Propositions and discussed how it can enhance feedback and accuracy in applications such as educational assessment. Second, we described an effective method for automatically extracting MMPs. The results of this approach were shown to be considerably higher than two meaningful baselines (0.184 absolute improvement on F_1 over a better baseline – 33% error reduction), validating the approach. Third, we successfully classified MMPs with respect to their importance, achieving a weighted F_1-score of 0.815 on the test set. This will enable applications, such as ITS, to respond appropriately based on different types of MMPs. This work introduces a new fully automated, domain-independent foundation for analyzing students' free responses, at a level of granularity that is appropriate for contextual comparison. The corpus described in this paper is publicly available for research purposes and represents a substantial contribution to multiple NLP sub-communities. This will quicken the pace of much related research. The annotated corpus along with the annotation guidelines will be available from the H*i*LT Lab Resources webpage.[2]

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.

[2] http://hilt.cse.unt.edu/resources.html

References

Stergos Afantenos, Pascal Denis, Philippe Muller, and Laurence Danlos. 2010. Learning recursive segments for discourse parsing. *arXiv preprint arXiv:1003.5372*.

Steven Burrows, Iryna Gurevych, and Benno Stein. 2015. The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25(1):60–117.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Douglas Duncan. 2006. Clickers: A new teaching aid with exceptional promise. *Astronomy Education Review*, 5(1):70–88.

Christiane Fellbaum. 1998. A semantic network of english verbs. *WordNet: An electronic lexical database*, 3:153–178.

Carmen Fies and Jill Marshall. 2006. Classroom response systems: A review of the literature. *Journal of Science Education and Technology*, 15(1):101–109.

Arthur C Graesser, Xiangen Hu, Suresh Susarla, Derek Harter, NK Person, Max Louwerse, Brent Olde, et al. 2001. Autotutor: An intelligent tutor and conversational tutoring scaffold. *10th ICAI in Education*, pages 47–49.

David Graff and C Cieri. 2003. English gigaword corpus. *Linguistic Data Consortium*.

Leonhard Hennig, Ernesto William De Luca, and Sahin Albayrak. 2010. Learning summary content units with topic modeling. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 391–399. Association for Computational Linguistics.

Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. 2010. A sequential model for discourse segmentation. In *Computational Linguistics and Intelligent Text Processing*, pages 315–326. Springer.

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

Maxim Makatchev, Pamela W Jordan, and Kurt VanLehn. 2004. Abductive theorem proving for analyzing student explanations to guide feedback in intelligent tutoring systems. *Journal of Automated Reasoning*, 32(3):187–226.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Tom Mitchell, Terry Russell, Peter Broomhead, and Nicola Aldridge. 2002. Towards robust computerised marking of free-text responses.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method.

Ani Nenkova, Rebecca Passonneau, and Kathleen McKeown. 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing (TSLP)*, 4(2):4.

Rodney d. Nielsen, Wayne Ward, and James h. Martin. 2009. Recognizing entailment in intelligent tutoring systems*. *Nat. Lang. Eng.*, 15(4):479–501, October.

Frank Paiva, James Glenn, Karen Mazidi, Robert Talbot, Ruth Wylie, Michelene TH Chi, Erik Dutilly, Brandon Helding, Mingyu Lin, Susan Trickett, et al. 2014. Comprehension seeding: Comprehension through self explanation, enhanced discussion, and inquiry generation. In *Intelligent Tutoring Systems*, pages 283–293. Springer.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Livia Polanyi, Chris Culy, Martin Van Den Berg, Gian Lorenzo Thione, and David Ahn. 2004. A rule based approach to discourse parsing. In *Proceedings of SIGDIAL*, volume 4.

Heather Pon-Barry, Brady Clark, Karl Schultz, Elizabeth Owen Bratt, and Stanley Peters. 2004. Contextualizing learning in a reflective conversational tutor. In *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on*, pages 236–240. IEEE.

Carolyn P Rosé, Antonio Roque, Dumisizwe Bhembe, and Kurt Vanlehn. 2003. A hybrid text classification approach for analysis of student essays. In *Proceedings of the HLT-NAACL 03 workshop on Building educational applications using natural language processing-Volume 2*, pages 68–75. Association for Computational Linguistics.

Karin Kipper Schuler. 2005. Verbnet: A broad-coverage, comprehensive verb lexicon.

Radu Soricut and Daniel Marcu. 2003. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 149–156. Association for Computational Linguistics.

Rajen Subba and Barbara Di Eugenio. 2007. Automatic discourse segmentation using neural networks. In *Proc. of the 11th Workshop on the Semantics and Pragmatics of Dialogue*, pages 189–190.

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

Jana Z Sukkarieh, Stephen G Pulman, and Nicholas Raikes. 2003. Auto-marking: using computational linguistics to score short, free text responses. In *th annual conference of the International Association for Educational Assessment (IAEA), Manchester, UK.*