A Hybrid Text Classification Approach for Analysis of Student Essays

Carolyn P. Rosé, Antonio Roque, Dumisizwe Bhembe, Kurt Vanlehn
Learning Research and Development Center, University of Pittsburgh,
3939 O'Hara St., Pittsburgh, PA 15260
rosecp,roque,bhembe,vanlehn@pitt.edu

Abstract

We present CarmelTC, a novel hybrid text classification approach for analyzing essay answers to qualitative physics questions, which builds upon work presented in (Rosé et al., 2002a). CarmelTC learns to classify units of text based on features extracted from a syntactic analysis of that text as well as on a Naive Bayes classification of that text. We explore the tradeoffs between symbolic and "bag of words" approaches. Our goal has been to combine the strengths of both of these approaches while avoiding some of the weaknesses. Our evaluation demonstrates that the hybrid CarmelTC approach outperforms two "bag of words" approaches, namely LSA and a Naive Bayes, as well as a purely symbolic approach.

1 Introduction

In this paper we describe CarmelTC, a novel hybrid text classification approach for analyzing essay answers to qualitative physics questions. In our evaluation we demonstrate that the novel hybrid CarmelTC approach outperforms both Latent Semantic Analysis (LSA) (Landauer et al., 1998; Laham, 1997) and Rainbow (McCallum, 1996; McCallum and Nigam, 1998), which is a Naive Bayes approach, as well as a purely symbolic approach similar to (Furnkranz et al., 1998). Whereas LSA and Rainbow are pure "bag of words" approaches, CarmelTC is a rule learning approach where rules for classifying units of text rely on features extracted from a syntactic analysis of that text as well as on a "bag of words" classification of that text. Thus, our evaluation demonstrates the advantage of combining predictions from symbolic and "bag of words" approaches for text classification. Similar to (Furnkranz et al., 1998), neither CarmelTC nor the purely symbolic approach require any domain specific knowledge engineering or text annotation beyond providing a training corpus of texts matched with appropriate classifications, which is also necessary for Rainbow, and to a much lesser extent for LSA.

CarmelTC was developed for use inside of the Why2-Atlas conceptual physics tutoring system (VanLehn et al., 2002; Graesser et al., 2002) for the purpose of grading short essays written in response to questions such as "Suppose you are running in a straight line at constant speed. You throw a pumpkin straight up. Where will it land? Explain." This is an appropriate task domain for pursuing questions about the benefits of tutorial dialogue for learning because questions like this one are known to elicit robust, persistent misconceptions from students, such as "heavier objects exert more force." (Hake, 1998; Halloun and Hestenes, 1985). In Why2-Atlas, a student first types an essay answering a qualitative physics problem. A computer tutor then engages the student in a natural language dialogue to provide feedback, correct misconceptions, and to elicit more complete explanations. The first version of Why2-Atlas was deployed and evaluated with undergraduate students in the spring of 2002; the system is continuing to be actively developed (Graesser et al., 2002).

In contrast to many previous approaches to automated essay grading (Burstein et al., 1998; Foltz et al., 1998; Larkey, 1998), our goal is not to assign a letter grade to student essays. Instead, our purpose is to tally which set of "correct answer aspects" are present in student essays. For example, we expect satisfactory answers to the example question above to include a detailed explanation of how Newton's first law applies to this scenario. From Newton's first law, the student should infer that the pumpkin and the man will continue at the same constant horizontal velocity that they both had before the release. Thus, they will always have the same displacement from

the point of release. Therefore, after the pumpkin rises and falls, it will land back in the man's hands. Our goal is to coach students through the process of constructing good physics explanations. Thus, our focus is on the physics content and not the quality of the student's writing, in contrast to (Burstein et al., 2001).

2 Student Essay Analysis

We cast the Student Essay Analysis problem as a text classification problem where we classify each sentence in the student's essay as an expression one of a set of "correct answer aspects", or "nothing" in the case where no "correct answer aspect" was expressed.

After a student attempts an initial answer to the question, the system analyzes the student's essay to assess which key points are missing from the student's argument. The system then uses its analysis of the student's essay to determine which help to offer that student. In order to do an effective job at selecting appropriate interventions for helping students improve their explanations, the system must perform a highly accurate analysis of the student's essay. Identifying key points as present in essays when they are not (i.e., false alarms), cause the system to miss opportunities to help students improve their essays. On the other hand, failing to identify key points that are indeed present in student essays causes the system to offer help where it is not needed, which can frustrate and even confuse students. A highly accurate inventory of the content of student essays is required in order to avoid missing opportunities to offer needed instruction and to avoid offering inappropriate feedback, especially as the completeness of student essays increases (Rosé et al., 2002a; Rosé et al., 2002c).

In order to compute which set of key points, i.e., "correct answer aspects", are included in a student essay, we first segment the essay at sentence boundaries. Note that run-on sentences are broken up. Once an essay is segmented, each segment is classified as corresponding to one of the set of key points or "nothing" if it does not include any key point. We then take an inventory of the classifications other than "nothing" that were assigned to at least one segment. Thus, our approach is similar in spirit to that taken in the AUTO-TUTOR system (Wiemer-Hastings et al., 1998), where Latent Semantic Analysis (LSA) (Landauer et al., 1998; Laham, 1997) was used to tally which subset of "correct answer aspects" students included in their natural language responses to short essay questions about computer literacy.

We performed our evaluation over essays collected from students interacting with our tutoring system in response to the question "Suppose you are running in a straight line at constant speed. You throw a pumpkin straight up. Where will it land? Explain.", which we refer to as the Pumpkin Problem. Thus, there are a total of six alternative classifications for each segment:

Class 1 Sentence expresses the idea that after the release the only force acting on the pumpkin is the downward force of gravity.

Class 2 Sentence expresses the idea that the pumpkin continues to have a constant horizontal velocity after it is released.

Class 3 Sentence expresses the idea that the horizontal velocity of the pumpkin continues to be equal to the horizontal velocity of the man.

Class 4 Sentence expresses the idea that the pumpkin and runner cover the same distance over the same time.

Class 5 Sentence expresses the idea that the pumpkin will land on the runner.

Class 6 Sentence does not adequately express any of the above specified key points.

Note that this classification task is strikingly different from those typically used for evaluating text classification systems. First, these classifications represent specific whole propositions rather than general topics, such as those used for classifying web pages (Craven et al., 1998), namely "student", "faculty", "staff", etc. Secondly, the texts are much shorter, i.e., one sentence in comparison with a whole web page, which is a disadvantage for "bag of words" approaches.

In some cases what distinguishes sentences from one class and sentences from another class is very subtle. For example, "Thus, the pumpkin's horizontal velocity, which is equal to that of the man when he released it, will remain constant." belongs to Class 2 although it could easily be mistaken for Class 3. Similarly, "So long as no other horizontal force acts upon the pumpkin while it is in the air, this velocity will stay the same.", belongs to Class 2 although looks similar on the surface to either Class 1 or 3. A related problem is that sentences that should be classified as "nothing" may look very similar on the surface to sentences belonging to one or more of the other classes. For example, "It will land on the ground where the runner threw it up." contains all of the words required to correctly express the idea corresponding to Class 5, although it does not express this idea, and in fact expresses a wrong idea. These very subtle distinctions also pose problems for "bag of words" approaches since they base their decisions only on which words are present regardless of their order or the functional relationships between them. That might suggest that a symbolic approach involving syntactic and semantic interpretation

might be more successful. However, while symbolic approaches can be more precise than "bag of words" approaches, they are also more brittle. And approaches that rely both on syntactic and semantic interpretation require a larger knowledge engineering effort as well.

3 CarmelTC

Figure 1: **This example shows the deep syntactic parse of a sentence.**

Sentence: The pumpkin moves slower because the man is not exerting a force on it.

Deep Syntactic Analysis

```
((clause2
 ((mood *declarative)
  (root move)
  (tense present)
  (subj
     ((cat dp)(root pumpkin)
      (specifier ((cat detp)(def +)(root the)))
      (modifier ((car adv) (root slow)))))))
 (clause2
  (mood *declarative)
  (root exert)
  (tense present)
  (negation +)
  (causesubj
     ((cat dp)(root man)(agr 3s)
      (specifier
          ((cat detp)(def +)(root the)))))
  (subj
     ((cat dp)(root force)
      (specifier ((cat detp)(root a)))))
  (obj ((cat dp)(root it))))
 (connective because))
```

The hybrid CarmelTC approach induces decision trees using features from both a deep syntactic functional analysis of an input text as well as a prediction from the Rainbow Naive Bayes text classifier (McCallum, 1996; McCallum and Nigam, 1998) to make a prediction about the correct classification of a sentence. In addition, it uses features that indicate the presence or absence of words found in the training examples. Since the Naive Bayes classification of a sentence is more informative than any single one of the other features provided, CarmelTC can be conceptualized as using the other features to decide whether or not to believe the Naive Bayes classification, and if not, what to believe instead.

From the deep syntactic analysis of a sentence, we extract individual features that encode functional relationships between syntactic heads (e.g., (subj-throw man)), tense information (e.g., (tense-throw past)), and information about passivization and negation (e.g., (negation-throw +) or (passive-throw -)). See Figures 1 and 2. Rainbow has been used for a wide range of text classification tasks. With Rainbow, P(doc,Class), i.e., the probability of a document belonging to class Class, is estimated by multiplying P(Class), i.e., the prior probability of the class, by the product over all of the words w_i found in the text of $P(w_I|Class)$, i.e., the probability of the word given that class. This product is normalized over the prior probability of all words. Using the individual features extracted from the deep syntactic analysis of the input as well as the "bag of words" Naive Bayes classification of the input sentence, CarmelTC builds a vector representation of each input sentence, with each vector position corresponding to one of these features. We then use the ID3 decision tree learning algorithm (Mitchell, 1997; Quinlin, 1993) to induce rules for identifying sentence classes based on these feature vectors.

Figure 2: **This example shows the features extracted from the deep syntactic parse of a sentence.**

Sentence: The pumpkin moves slower because the man is not exerting a force on it.

Extracted Features

```
(tense-move present)
(subj-move pumpkin)
(specifier-pumpkin the)
(modifier-move slow)
(tense-exert present)
(negation-exert +)
(causesubj-exert man)
(subj-exert force)
(obj-exert it)
(specifier-force a)
(specifier-man the)
```

The symbolic features used for the CarmelTC approach are extracted from a deep syntactic functional analysis constructed using the CARMEL broad coverage English syntactic parsing grammar (Rosé, 2000) and the large scale COMLEX lexicon (Grishman et al., 1994), containing 40,000 lexical items. For parsing we use an incremental version of the LCFLEX robust parser (Rosé et al., 2002b; Rosé and Lavie, 2001), which was designed for efficient, robust interpretation. While computing a deep syntactic analysis is more computationally expensive than computing a shallow syntactic analysis, we can do so very efficiently using the incrementalized version of LCFLEX because it takes advantage of student typing time to reduce the time delay between when students

submit their essays and when the system is prepared to respond.

Syntactic feature structures produced by the CARMEL grammar factor out those aspects of syntax that modify the surface realization of a sentence but do not change its deep functional analysis. These aspects include tense, negation, mood, modality, and syntactic transformations such as passivization and extraction. In order to do this reliably, the component of the grammar that performs the deep syntactic analysis of verb argument functional relationships was generated automatically from a feature representation for each of COMLEX's verb subcategorization tags. It was verified that the 91 verb subcategorization tags documented in the COMLEX manual were covered by the encodings, and thus by the resulting grammar rules. These tags cover a wide range of patterns of syntactic control and predication relationships. Each tag corresponds to one or more case frames. Each case frame corresponds to a number of different surface realizations due to passivization, relative clause extraction, and wh-movement. Altogether there are 519 syntactic patterns covered by the 91 subcategorization tags, all of which are covered by the grammar.

There are nine syntactic functional roles assigned by the grammar. These roles include subj (subject), causesubj (causative subject), obj (object), iobj (indirect object), pred (descriptive predicate, like an adjectival phrase or an adverb phrase), comp (a clausal complement), modifier, and possessor. The roles pertaining to the relationship between a verb and its arguments are assigned based on the subcat tags associated with verbs in COMLEX. However, in some cases, arguments that COMLEX assigns the role of subject get redefined as causesubj (causative subject). For example, the subject in "the pumpkin moved" is just a subject but in "the man moved the pumpkin", the subject would get the role causesubj instead since 'move' is a causative-inchoative verb and the obj role is filled in in the second case [1]. The modifier role is used to specify the relationship between any syntactic head and its adjunct modifiers. Possessor is used to describe the relationship between a head noun and its genitive specifier, as in man in either "the man's pumpkin" or "the pumpkin of the man".

With the hybrid CarmelTC approach, our goal has been to keep as many of the advantages of both symbolic analysis as well as "bag of words" classification approaches as possible while avoiding some of the pitfalls of each. Since the CarmelTC approach does not use the syntactic analysis as a whole, it does not require that the system be able to construct a totally complete and correct syntactic analysis of the student's text input. It can very effectively make use of partial parses. Thus, it is more robust than purely symbolic approaches where decisions are based on complete analyses of texts. And since it makes use only of the syntactic analysis of a sentence, rather than also making use of a semantic interpretation, it does not require any sort of domain specific knowledge engineering. And yet the syntactic features provide information normally not available to "bag of words" approaches, such as functional relationships between syntactic heads and scope of negation and other types of modifiers.

[1]The causative-inchoative verb feature is one that we added to verb entries in COMLEX, not one of the features provided by the lexicon originally.

4 Related Work: Combining Symbolic and Bag of Words Approaches

CarmelTC is most similar to the text classification approach described in (Furnkranz et al., 1998). In the approach described in (Furnkranz et al., 1998), features that note the presence or absence of a word from a text as well as extraction patterns from AUTOSLOG-TS (Riloff, 1996) form the feature set that are input to the RIPPER (Cohen, 1995), which learns rules for classifying texts based on these features. CarmelTC is similar in spirit in terms of both the sorts of features used as well as the general sort of learning approach. However, CarmelTC is different from (Furnkranz et al., 1998) in several respects.

Where (Furnkranz et al., 1998) make use of AUTOSLOG-TS extraction patterns, CarmelTC makes use of features extracted from a deep syntactic analysis of the text. Since AUTOSLOG-TS performs a surface syntactic analysis, it would assign a different representation to all aspects of these texts where there is variation in the surface syntax. Thus, the syntactic features extracted from our syntactic analyses are more general. For example, for the sentence "The force was applied by the man to the object", our grammar assigns the same functional roles as for "The man applied the force to the object" and also for the noun phrase "the man that applied the force to the object". This would not be the case for AUTOSLOG-TS.

Like (Furnkranz et al., 1998), we also extract word features that indicate the presence or absence of a root form of a word from the text. However, in contrast for CarmelTC one of the features for each training text that is made available to the rule learning algorithm is the classification obtained using the Rainbow Naive Bayes classifier (McCallum, 1996; McCallum and Nigam, 1998).

Because the texts classified with CarmelTC are so much shorter than those of (Furnkranz et al., 1998), the feature set provided to the learning algorithm was small enough that it was not necessary to use a learning algorithm as sophisticated as RIPPER (Cohen, 1995). Thus, we used ID3 (Mitchell, 1997; Quinlin, 1993) instead with excellent results. Note that in contrast to CarmelTC, the (Furnkranz et al., 1998) approach is purely symbolic.

Thus, all of its features are either word level features or surface syntactic features.

Recent work has demonstrated that combining multiple predictors yields combined predictors that are superior to the individual predictors in cases where the individual predictors have complementary strengths and weaknesses (Larkey and Croft, 1996; Larkey and Croft, 1995). We have argued that this is the case with symbolic and "bag of words" approaches. Thus, we have reason to expect a hybrid approach that makes a prediction based on a combination of these single approaches would yield better results than either of these approaches alone. Our results presented in Section 5 demonstrate that this is true.

Other recent work has demonstrated that symbolic and "Bag of Words" approaches can be productively combined. For example, syntactic information can be used to modify the LSA space of a verb in order to make LSA sensitive to different word senses (Kintsch, 2002). However, this approach has only been applied to the analysis of mono-transitive verbs. Furthermore, it has never been demonstrated to improve LSA's effectiveness at classifying texts.

In the alternative Structured Latent Semantic Analysis (SLSA) approach, hand-coded subject-predicate information was used to improve the results obtained by LSA for text classification (Wiemer-Hastings and Zipitria, 2001), but no fully automated evaluation of this approach has been published.

In contrast to these two approaches, CarmelTC is both fully automatic, in that the symbolic features it uses are obtained without any hand coding whatsoever, and fully general, in that it applies to the full range of verb subcategorization frames covered by the COMLEX lexicon, not only mono-transitive verbs. In Section 5 we demonstrate that CarmelTC outperforms both LSA and Rainbow, two alternative bag of words approaches, on the task of student essay analysis.

5 Evaluation

We conducted an evaluation to compare the effectiveness of CarmelTC at analyzing student essays in comparison to LSA, Rainbow, and a purely symbolic approach similar to (Furnkranz et al., 1998), which we refer to here as CarmelTCsymb. CarmelTCsymb is identical to CarmelTC except that it does not include in its feature set the prediction from Rainbow. Thus, by comparing CarmelTC with Rainbow and LSA, we can demonstrate the superiority of our hybrid approach to purely "bag of words" approaches. And by comparing with CarmelTCsymb, we can demonstrate the superiority of our hybrid approach to an otherwise equivalent purely symbolic approach.

We conducted our evaluation over a corpus of 126 previously unseen student essays in response to the Pumpkin Problem described above, with a total of 500 text segments, and just under 6000 words altogether. We first tested to see if the text segments could be reliably tagged by humans with the six possible Classes associated with the problem. Note that this includes "nothing" as a class, i.e., Class 6. Three human coders hand classified text segments for 20 essays. We computed a pairwise Kappa coefficient (Cohen, 1960) to measure the agreement between coders, which was always greater than .75, thus demonstrating good agreement according to the Krippendorf scale (Krippendorf, 1980). We then selected two coders to individually classify the remaining sentences in the corpus. They then met to come to a consensus on the tagging. The resulting consensus tagged corpus was used as a gold standard for this evaluation. Using this gold standard, we conducted a comparison of the four approaches on the problem of tallying the set of "correct answer aspects" present in each student essay.

The LSA space used for this evaluation was trained over three first year physics text books. The other three approaches are trained over a corpus of tagged examples using a 50 fold random sampling evaluation, similar to a cross-validation methodology. On each iteration, we randomly selected a subset of essays such that the number of text segments included in the test set were greater than 10 but less than 15. The randomly selected essays were then used as a test set for that iteration, and the remainder of the essays were used for training in addition to a corpus of 248 hand tagged example sentences extracted from a corpus of human-human tutoring transcripts in our domain. The training of the three approaches differed only in terms of how the training data was partitioned. Rainbow and CarmelTCsymb were trained using all of the example sentences in the corpus as a single training set. CarmelTC, on the other hand, required partitioning the training data into two subsets, one for training the Rainbow model used for generating the value of its Rainbow feature, and one subset for training the decision trees. This is because for CarmelTC, the data for training Rainbow must be separate from that used to train the decision trees so the decision trees are trained from a realistic distribution of assigned Rainbow classes based on its performance on unseen data rather than on Rainbow's training data.

In setting up our evaluation, we made it our goal to present our competing approaches in the best possible light in order to provide CarmelTC with the strongest competitors as possible. Note that LSA works by using its trained LSA space to construct a vector representation for any text based on the set of words included therein. It can thus be used for text classification by comparing the vector obtained for a set of exemplar texts for each class with that obtained from the text to be classified. We tested LSA using as exemplars the same set of examples used

Figure 3: **This Table compares the performance of the 4 alternative approaches in the per essay evaluation in terms of precision, recall, false alarm rate, and f-score.**

Approach	Precision	Recall	False Alarm Rate	F-Score
LSA	93%	54%	3%	.70
Rainbow	81%	73%	9%	.77
CarmelTCsymb	88%	72%	7%	.79
CarmelTC	90%	80%	8%	.85

as Rainbow training data, but it always performed better when using a small set of hand picked exemplars. Thus, we present results here using only those hand picked exemplars. For every approach except LSA, we first segmented the essays at sentence boundaries and classified each sentence separately. However, for LSA, rather than classify each segment separately, we compared the LSA vector for the entire essay to the exemplars for each class (other than "nothing"), since LSA's performance is better with longer texts. We verified that LSA also performed better specifically on our task under these circumstances. Thus, we compared each essay to each exemplar, and we counted LSA as identifying the corresponding "correct answer aspect" if the cosine value obtained by comparing the two vectors was above a threshold. We tested LSA with threshold values between .1 and .9 at increments of .1 as well as testing a threshold of .53 as is used in the AUTO-TUTOR system (Wiemer-Hastings et al., 1998). As expected, as the threshold increases from .1 to .9, recall and false alarm rate both decrease together as precision increases. We determined based on computing f-scores[2] for each threshold level that .53 achieves the best trade off between precision and recall. Thus, we used a threshold of .53, to determine whether LSA identified the corresponding key point in the student essay or not for the evaluation presented here.

We evaluated the four approaches in terms of precision, recall, false alarm rate, and f-score, which were computed for each approach for each test essay, and then averaged over the whole set of test essays. We computed precision by dividing the number of "correct answer aspects" (CAAs) correctly identified by the total number of CAAs identified[3] We computed recall by dividing the number of CAAs correctly identified over the number of CAAs actually present in the essay[4] False alarm rate was computed by dividing the number of CAAs incorrectly identified by the total number of CAAs that could potentially be incorrectly identified[5]. F-scores were computed using 1 as the beta value in order to treat precision and recall as equally important.

The results presented in Figure 3 clearly demonstrate that CarmelTC outperforms the other approaches. In particular, CarmelTC achieves the highest f-score, which combines the precision and recall scores into a single measure. In comparison with CarmelTCsymb, CarmelTC achieves a higher recall as well as a slightly higher precision. While LSA achieves a slightly higher precision, its recall is much lower. Thus, the difference between the two approaches is clearly shown in the f-score value, which strongly favors CarmelTC. Rainbow achieves a lower score than CarmelTC in terms of precision, recall, false alarm rate, and f-score.

6 Conclusion and Current Directions

In this paper we have introduced the CarmelTC text classification approach as it is applied to the problem of student essay analysis in the context of a conceptual physics tutoring system. We have evaluated CarmelTC over data collected from students interacting with our system in response to one of its 10 implemented conceptual physics problems. Our evaluation demonstrates that the novel hybrid CarmelTC approach outperforms both Latent Semantic Analysis (LSA) (Landauer et al., 1998; Laham, 1997) and a Naive Bayes approach (McCallum, 1996; McCallum and Nigam, 1998) as well as a purely symbolic approach similar to (Furnkranz et al., 1998). We plan to run a larger evaluation with essays from multiple problems to test the generality of our result. We also plan to experiment with other rule learning approaches, such as RIPPER (Cohen, 1995).

7 Acknowledgments

This research was supported by the Office of Naval Research, Cognitive Science Division under grant number N00014-0-1-0600 and by NSF grant number 9720359 to CIRCLE, Center for Interdisciplinary Research on Constructive Learning Environments at the University of Pittsburgh and Carnegie Mellon University.

[2]We computed our f-scores with a beta value of 1 in order to treat precision and recall as equally important.

[3]For essays containing no CAAs, we counted precision as 1 if none were identified and 0 otherwise.

[4]For essays with no CAAs present, we counted recall as 1 for all approaches.

[5]For essays containing all possible CAAs, false alarm rate was counted as 0 for all approaches.

References

J. Burstein, K. Kukich, S. Wolff, C. Lu, M. Chodorow, L. Braden-Harder, and M. D. Harris. 1998. Automated scoring using a hybrid feature identification technique. In *Proceedings of COLING-ACL'98*, pages 206–210.

J. Burstein, D. Marcu, S. Andreyev, and M. Chodorow. 2001. Towards automatic classification of discourse elements in essays. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics, Toulouse, France.*

J. Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(Winter):37–46.

W. W. Cohen. 1995. Fast effective rule induction. In *Proceedings of the 12th International Conference on Machine Learning*.

M. Craven, D. DiPasquio, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. 1998. Learning to extract symbolic knowledge from the world wide web. In *Proceedings of the 15th National Conference on Artificial Intelligence*.

P. W. Foltz, W. Kintsch, and T. Landauer. 1998. The measurement of textual coherence with latent semantic analysis. *Discourse Processes*, 25(2-3):285–307.

J. Furnkranz, T. Mitchell Mitchell, and E. Riloff. 1998. A case study in using linguistic phrases for text categorization on the www. In *Proceedings from the AAAI/ICML Workshop on Learning for Text Categorization*.

A. Graesser, K. Vanlehn, TRG, and NLT Group. 2002. Why2 report: Evaluation of why/atlas, why/autotutor, and accomplished human tutors on learning gains for qualitative physics problems and explanations. Technical report, LRDC Tech Report, University of Pittsburgh.

R. Grishman, C. Macleod, and A. Meyers. 1994. COMLEX syntax: Building a computational lexicon. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING-94)*.

R. R. Hake. 1998. Interactive-engagement versus traditional methods: A six-thousand student survey of mechanics test data for introductory physics students. *American Journal of Physics*, 66(64).

I. A. Halloun and D. Hestenes. 1985. The initial knowledge state of college physics students. *American Journal of Physics*, 53(11):1043–1055.

W. Kintsch. 2001. Predication. *Cognitive Science*, 25:173–202.

K. Krippendorf. 1980. *Content Analysis: An Introduction to Its Methodology*. Sage Publications.

D. Laham. 1997. Latent semantic analysis approaches to categorization. In *Proceedings of the Cognitive Science Society*.

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

L. S. Larkey and W. B. Croft. 1995. Automatic assignment of icd9 codes to discharge summaries. Technical Report IR-64, University of Massachusetts Center for Intelligent Information Retrieval.

L. S. Larkey and W. B. Croft. 1996. Combining classifiers in text categorization. In *Proceedings of SIGIR*.

L. Larkey. 1998. Automatic essay grading using text categorization techniques. In *Proceedings of SIGIR*.

A. McCallum and K. Nigam. 1998. A comparison of event models for naive bayes text classification. In *Proceedings of the AAAI-98 Workshop on Learning for Text Classification*.

Andrew Kachites McCallum. 1996. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. http://www.cs.cmu.edu/ mccallum/bow.

T. M. Mitchell. 1997. *Machine Learning*. McGraw Hill.

J. R. Quinlin. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers: San Mateo, CA.

E. Riloff. 1996. Using learned extraction patterns for text classification. In S. Wermter, R. Riloff, and G. Scheler, editors, *Connectionist, Statistical, and Symbolic Approaches for Natural Language Processing*. Springer-Verlag.

C. P. Rosé and A. Lavie. 2001. Balancing robustness and efficiency in unification augmented context-free parsers for large practical applications. In J. C. Junqua and G. Van Noord, editors, *Robustness in Language and Speech Technologies*. Kluwer Academic Press.

C. P. Rosé, D. Bhembe, A. Roque, S. Siler, R. Srivastava, and K. Vanlehn. 2002a. A hybrid language understanding approach for robust selection of tutoring goals. In *Proceedings of the Intelligent Tutoring Systems Conference*.

C. P. Rosé, D. Bhembe, A. Roque, and K. VanLehn. 2002b. An efficient incremental architecture for robust interpretation. In *Proceedings of the Human Languages Technology Conference*, pages 307–312.

C. P. Rosé, P. Jordan, and K. VanLehn. 2002c. Can we help students with high initial competency? In *Proceedings of the ITS Workshop on Empirical Methods for Tutorial Dialogue Systems*.

C. P. Rosé. 2000. A framework for robust semantic interpretation. In *Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 311–318.

K. VanLehn, P. Jordan, C. P. Rosé, and The Natural Languag e Tutoring Group. 2002. The architecture of why2-atlas: a coach for qualitative physics essay writing. In *Proceedings of the Intelligent Tutoring Systems Conference*, pages 159–167.

P. Wiemer-Hastings and I. Zipitria. 2001. Rules for syntax, vectors for semantics. In *Proceedings of the Twenty-third Annual Conference of the Cognitive Science Society*.

P. Wiemer-Hastings, A. Graesser, D. Harter, and the Tutoring Res earch Group. 1998. The foundations and architecture of autotutor. In B. Goettl, H. Halff, C. Redfield, and V. Shute, editors, *Intelligent Tutoring Systems: 4th International Conference (ITS '98)*, pages 334–343. Springer Verlag.