Combining Competing Language Understanding Approaches in an Intelligent Tutoring System

Pamela W. Jordan, Maxim Makatchev, and Kurt VanLehn

Learning Research and Development Center, Intelligent Systems Program and Computer Science Department, University of Pittsburgh, Pittsburgh PA 15260
{pjordan,maxim,vanlehn}@pitt.edu

Abstract. When implementing a tutoring system that attempts a deep understanding of students' natural language explanations, there are three basic approaches to choose between; symbolic, in which sentence strings are parsed using a lexicon and grammar; statistical, in which a corpus is used to train a text classifier; and hybrid, in which rich, symbolically produced features supplement statistical training. Because each type of approach requires different amounts of domain knowledge preparation and provides different quality output for the same input, we describe a method for heuristically combining multiple natural language understanding approaches in an attempt to use each to its best advantage. We explore two basic models for combining approaches in the context of a tutoring system; one where heuristics select the first satisficing representation and another in which heuristics select the highest ranked representation.

1 Introduction

Implementing an intelligent tutoring system that attempts a deep understanding of a student's natural language (NL) explanation is a challenging and time consuming undertaking even when making use of existing NL processing tools and techniques [1–3]. A motivation for attempting a deep understanding of an explanation is so that a tutoring system can reason about the domain knowledge expressed in the student's explanation in order to diagnose errors that are only implicitly expressed [4] and to provide substantive feedback that encourages further self-explanation [5]. To accomplish these tutoring system tasks, the NL technology must be able to map typical student language to an appropriate domain level representation language. While some NL mapping approaches require relatively little domain knowledge preparation there is currently still a trade-off with the quality of the representation produced especially as the complexity of the representation language increases.

Although most NL mapping approaches have been rigorously evaluated, the results may not scale-up or generalize to the tutoring system domain. First it may not be practical to carefully prepare large amounts of domain knowledge in the same manner as may have been done for the evaluation of an NL approach. This is especially a problem for tutoring systems since they need to cover a large

amount of domain knowledge to have an impact on student learning. Second, acceptable performance results may vary across applications if the requirements for representation fidelity vary. For example, a document retrieval application may not require a deep understanding of every sentence in the document to be successful whereas providing tutorial feedback to students on the content of what they write may. Finally, while one approach may be more promising than another for providing a better quality representation, the time required to prepare the domain knowledge to achieve the desired fidelity is not yet reliably predictable. For these reasons, it may be advisable to include multiple approaches and to re-examine how the approaches are integrated within the tutoring system as the domain coverage expands and improves over time.

Our goal in this paper is to examine ways in which multiple language mapping approaches can be integrated within one tutoring system so that each approach is used to its best advantage relative to a particular time-slice in the life-cycle of the knowledge development for the tutoring system. At a given time-slice, one approach may be functioning better than another but we must anticipate that the performances may change when there is a significant change in the domain knowledge provided. Our approach for integrating multiple mapping approaches, each with separate evolving knowledge sources, is to set up a competition between them and allow a deliberative process to decide for every student sentence processed which representation is the best one to use. This approach is similar to what is done in multi-agent architectures [6]. We will experimentally explore a variety of ways of competitively combining three types of NL understanding approaches in the context of the Why2-Atlas tutoring system; 1) symbolic, in which sentence strings are parsed using an NL lexicon and grammar 2) statistical, in which a corpus is used to train a text classifier and 3) hybrid, in which rich symbolic features are used to supplement the training of a text classifier.

First we will describe the Why2-Atlas tutoring domain and representation language to give an impression of the difficulty of the NL mapping task. Next we will characterize the expected performance differences of the individual approaches. Next we will describe how we measure performance and discuss how to go about selecting the best configuration for a particular knowledge development time-slice. Next we will describe two types of competition models and their selection heuristics where the heuristics evaluate representations relative to typical (but generally stated) representation failings we anticipate and have observed for each approach. Finally, we will examine the performance differences for various ways of combining the NL understanding approaches and compare them to two baselines; the current best single approach and tutoring on all possible topics.

2 Overview of the Why2-Atlas Domain and Representation Language

The Why2-Atlas system covers 5 qualitative physics problems on introductory mechanics. For each problem the student is expected to type an answer and explanation which the system analyzes in order to identify appropriate elicita-

Table 1. Slots for one body vector quantities with examples of slot filler constants.

Description	Slot sorts (examples of slot filler constants)
quantity	`Quantity1b` (velocity, acceleration)
identifier	`Id` (ID100)
body (or two bodies in case of `force`)	`Body` (pumpkin, man)
axial component or not	`Comp` (horizontal, vertical)
qualitative derivative of the magnitude	`D-mag` (constant, increase, decrease)
quantitative derivative of the magnitude	`D-mag-num` (*none*)
zero or non-zero magnitude	`Mag-zero` (zero, nonzero)
quantitative magnitude	`Mag-num` (*none*)
sign for axial component	`Dir` (pos,neg)
qualitative derivative of the direction	`D-dir` (constant, nonconstant)
beginning of time interval	`Time` (*problem specific*)
end of time interval	`Time` (*problem specific*)

tion, clarification and remediation tutoring goals. The details of the Why2-Atlas system are described in [1] and only the mapping of an isolated NL sentence to the Why2-Atlas representation language will be addressed in this paper. In this section we give an overview of the rich domain representation language that the system uses to support diagnosis and feedback.

The Why2-Atlas ontology is strongly influenced by previous qualitative physics reasoning work, in particular [7], but makes appropriate simplifications given the subset of physics the system is addressing. The Why2-Atlas ontology comprises bodies, states, physical quantities, times and relations. The ontology and representation language are described in detail in [4].

For the sake of simplicity, most bodies in the Why2-Atlas ontology have the semantics of point-masses. Body constants are problem specific. For example the body constants for one problem covered by Why2-Atlas are `pumpkin` and `man`.

Individual bodies can be in *states* such as `freefall`. Being in a particular state implies respective restrictions on the forces applied on the body. There is also the special state of `contact` between two bodies where `attached` bodies can exert mutual forces and the positions of the two bodies are equal, `detached` bodies do not exert mutual forces, and `moving-contact` bodies can exert mutual forces but there is no conclusion on their relative positions. The latter type of contact is introduced to account for point-mass bodies that are capable of pushing/pulling each other for certain time intervals (a non-impact type of contact), for example the man pushing a pumpkin up.

Physical quantities are represented as one or two body vectors. The one body vector quantities are `position`, `displacement`, `velocity`, `acceleration`, and `total-force` and the only two body one in the Why2-Atlas ontology is `force`. The single body scalar quantities are `duration`, `mass`, and `distance`.

Every physical quantity has slots and respective restrictions on the sort of a slot filler as shown in Table 1, where examples of slot filler constants of the proper sorts are shown in parentheses. Note that the sorts `Id`, `D-mag`, and `D-mag-num`

do not have specific constants. These slots are used only for cross-referencing between different propositions.

Time instants are basic primitives in the Why2-Atlas ontology and a time interval is a pair (t_i, t_j) of instants. This definition of time intervals is sufficient for implementing the semantics of *open time intervals* in the context of the mechanics domain.

Some of the multi-place relations in our domain are `before`, `rel-position` and `compare`. The relation `before` relates time instants in the obvious way. The relation `rel-position` provides the means to represent the relative position of two bodies with respect to each other, independently of the choice of a coordinate system—a common way to informally compare positions in NL. The relation `compare` is used to represent the ratio and difference of two quantities' magnitudes or for quantities that change over time, magnitudes of the derivatives.

The domain propositions are represented using order-sorted first-order logic (FOL) (see for example [8]). For example, "force of gravity acting on the pumpkin is constant and nonzero" has the following representation in which the generated *identifier* constants `f1` and `ph1` appear as arguments in the `due-to` relation predicate (sort information is omitted):

```
(force f1 ?body1 pumpkin ?comp constant ?d-mag-num nonzero ?mag-num ?dir
  ?d-dir ?t1 ?t2)
(due-to d1 f1 ph1)
(phenomenon ph1 gravity)
```

There is no explicit negation so a negative student statement such as "there is no force" is represented as the force being zero. The version of the system currently under development is extending the knowledge representation to cover disjunctions, conditional statements and other types of negations.

3 Overview of the language understanding approaches

In general, symbolic approaches are expected to yield good coverage and accuracy if sufficient knowledge of the domain can be captured and efficiently utilized. Whereas statistical and hybrid approaches are much easier to develop for a domain than symbolic ones and can provide just as good of coverage, those that use little more than a text corpus are expected to provide less accurate representations of what the student meant than pure symbolic approaches (once the knowledge engineering problem is adequately addressed).

Although there are many tools available for each type of approach, we developed Why2-Atlas domain knowledge sources for the symbolic approach Carmel [9], the statistical approach Rainbow [10] and the hybrid symbolic and statistical approach Rappel [11]. The knowledge development for each approach is still ongoing and at different levels of completeness, yet the system has been successfully used by students in two tutoring studies. Below we describe each of the approaches, as well as the tools we use, in more detail. We use the theoretical

strengths and weaknesses of each general type of approach as the basis for our hand-coded selection heuristics.

3.1 Symbolic Approach

The traditional approach for mapping NL to a knowledge representation language is symbolic; sentence strings are parsed using an NL lexicon and grammar. There are many practical and robust sentence-level syntactic parsers available for which wide coverage NL lexicons and grammars exist [12, 13, 9], but syntactic analysis can only canonicalize relative to syntactic aspects of lexical semantics [14]. For example, the similarity of "I baked a cake for her" and "I baked her a cake" is found but their similarity to "I made her a cake" is not.[1] The latter sort of canonicalization is typically provided by semantic analysis. But there is no general solution at this level because semantic analysis falls into the realm of cognition and mental representations [15] and must be engineered relative to the domain of interest.

CARMEL provides combined syntactic and semantic analysis using the LCFlex robust syntactic parser, a broad coverage grammar, and semantic constructor functions that are specific to the domain to be covered [9]. Given a specification of the desired representation language, it then maps the resulting analysis to the domain representation language. Until recently, semantic constructor functions had to be completely hand-generated for every lexical entry. Although tools to facilitate and expedite this level of knowledge representation are currently being developed [16, 17], it is still a significant knowledge engineering effort.

Because the necessary lexical-level knowledge engineering is difficult and time consuming and it is unclear how to predict when such a task will be sufficiently completed, there may be unexpected gaps in the semantic knowledge. Also robust parsing techniques can produce partial analyses and typically have a limited ability to self-evaluate the quality of the representation into which it maps a student sentence. So the ability to produce partial analyses in conjunction with gaps in the knowledge sources suggest that symbolic approaches will tend to undergenerate representations for sentences that weren't anticipated during the creation of their knowledge sources.

3.2 Statistical Approach

More recent approaches for processing NL are statistical; a corpus is used to train a wide variety of approaches for analyzing language. Statistical approaches are popular because there is relatively little effort involved to get such an approach working, if a representative corpus already exists. The most useful of these approaches for intelligent tutoring systems has been text classification in which a subtext is tagged as being a member of a particular class of interest and uses just the words in the class tagged corpus for training a classifier. This particular style

[1] The need to distinguish the semantic differences between "bake" and "made" depends on the application for which the representation will be used.

of classification is called a *bag of words* approach because the meaning that the organization of a sentence imparts is not considered. The classes themselves are generally expressed as text as well and are at the level of an exemplar of a text that is a member of the class. With this approach, the text can be mapped to its representation by looking up a hand-generated propositional representation for the exemplar text of the class identified at run-time.

RAINBOW is one such *bag of words* text classifier; in particular it is a Naive Bayes text classifier. The classes of interest must first be decided and then a training corpus developed where subtexts are annotated with the class to which it belongs. For the Why2-Atlas training, each sentence was annotated with one class. During training RAINBOW computes an estimate of the probability of a word in a particular class relative to the class labellings for the Why2-Atlas training sentences. Then when a new sentence is to be analyzed at run-time, RAINBOW calculates the posterior probabilities of each class relative to the words in the sentence and selects the class with the highest probability [10].

Like most statistical approaches, the quality of RAINBOW's analysis depends on the quality of its training data. Although good annotator agreement is possible for the classes of interest for the Why2-Atlas domain [18], we found the resulting training set for a class sometimes includes sentences that depend on a particular context for the full meaning of that class to be licensed. In practice the necessary context may not be present for the new sentence that is to be analyzed. This suggests that the statistical approach will tend to overgenerate representations. It is also possible for a student to express more than one key part of an explanation in a single sentence so that multiple class assignments would be more appropriate. This suggests that the statistical approach will also sometimes undergenerate since only the best classification is used. However, we expect the need for multiple class assignments to happen infrequently since the Why2-Atlas system includes a sentence segmenter that attempts to break up complex sentences before sentence understanding is attempted by any of the approaches.

3.3 Hybrid Approach

Finally, there are hybrids of symbolic and statistical approaches. For example, syntactic features can be used to supplement the training of a text classifier. Although the syntactic features often are obtained via statistical parsing methods, they are sometimes obtained via symbolic methods instead since the resulting feature set is richer [18]. With text classification, the classes are still generally defined via an exemplar of the class so the desired propositional representation must still be obtained via a look-up according to the class identified at run-time.

RAPPEL is a hybrid approach that uses symbolically-derived syntactic dependency features (obtained via MINIPAR [13, 19]) to train for classes that are defined at the representation language level [11] instead of at an informal text level. There is a separate classifier for each type of proposition in the knowledge representation language. Each classifier indicates whether a proposition of the type it recognizes is present and if so, which class it is. The class indicates

which slots are filled with which slot constants. There is then a one-to-one correspondence between a class and a proposition in the representation language. To arrive at the representation for a single sentence, RAPPEL applies all of the trained classifiers and then combines their results during a post-processing stage.

For Why2-Atlas we trained separate classifiers for every physics quantity, relation and state for a total of 27 different classifiers. For example, there is a separate classifier for `velocity` and another for `acceleration`. Bodies are also handled by separate classifiers; one for one body propositions and another for two body propositions. The basic approach for the body classifiers is similar to that used in statistical approaches to reference resolution (e.g. [20, 21]). The number of classes within each classifier depend on the number of slot constant filler combinations possible. For example, the class *v_h* encodes the proposition (velocity id1 horizontal ?body ?var$_1$ ... ?var$_n$) and the class *v_hip* encodes the proposition (velocity id2 horizontal ?body increase ?mag-zero ?mag-num pos ?t1 ?t2) where *v* represents the predicate `velocity`, *h* represents the slot constant `horizontal`, *i* represents the slot constant `increase` and *p* represents the constant `pos`.

Having a large number of classifiers and classes requires a larger, more comprehensive set of training data than is needed for a typical text classification approach. And just as with the preparation of the training data for the statistical approach, the annotator may still be influenced by the context of a sentence. However, we expect the impact of contextual dependencies to be less severe since the representation-defined classes are more formal and finer-grained than text-defined classes. For example, annotators may still resolve intersentential anaphora and ellipsis but the content related inferences needed to select a class are much finer-grained and therefore a closer fit to the actual meaning of the sentence.

Although we have classifiers and classes defined that cover the entire Why2-Atlas representation language, we have not yet provided training for the full representation language. Given the strong dependence of this approach on the completeness of the training data, we expect this approach to sometimes undergenerate just as an incomplete symbolic approach would and sometimes to overgenerate because of overgeneralizations during learning, just as with any statistical approach.

4 Computing and Comparing Performances

To measure the overall performance of the Why2-Atlas system when using different understanding approach configurations, we use a test suite of 35 held-out multi-sentence student essays (235 sentences total) that are annotated for the elicitation and remediation topics that are to be discussed with the student. Elicitation topics are tagged when prescribed, critical physics principles are missing from the student's explanation and remediation topics are tagged when the essay implicitly or explicitly exhibits any of a small number of misconceptions or errors that are typical of beginning students. From a language analysis perspec-

tive, the representation of the essay must be accurate enough to detect when physics principles are both properly and improperly expressed in the essay.

For the entire test suite we compute the number of true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) for the elicitation topics selected by the system relative to the elicitation topics annotated for the test suite essays. From this we compute recall = TP/(TP+FN), precision = TP/(TP+FP), and false alarm rate = FP/(FP+TN).

As a baseline measure, we compute the recall, precision and false alarm rate that results if all possible elicitations for a physics problem are selected. For our 35 essay test suite the recall is 1, precision is .61 and false alarm rate is 1. Although NL evaluations compute an F-measure (the harmonic average of recall and precision) in order to arrive at one number for comparing approaches, it does not allow errors to be considered as fully as with other analysis methods such as receiver operating characteristics (ROC) areas [22] and d′ [23]. These measures are similar in that they combine the recall and the false alarm rates into one number but allow for error skewing [22]. Rather than undertaking a full comparison of the various NL understanding approach configurations for this paper, we will instead look for those combinations that result in a high recall and a low false alarm rate. Error skewing depends on what costs we need to attribute to false negatives and false positives. Both potentially have negative impacts on student learning in that the former leaves out important information that should have been brought to the student's attention and the latter can confuse the student or cause lack of confidence in the system.

5 The Selection Heuristics

Although an NL understanding approach is not strictly an agent in the sense of [24] (e.g. it doesn't reason about goals or other agents) it can be treated architecturally as a service agent in the sense of [25] as has been done in many dialogue systems (e.g. [26, 3]). Generally the service agents supply slightly different information or are relevant in slightly different contexts so that the evaluator or coordinator decides which single service agent will be assigned a particular task. For example, [26] describes a system architecture that includes competing discourse strategy service agents and an evaluator that rates the competing strategies and selects the highest rated strategy agent to perform the communication task.

However, in the case of competing NL understanding approaches, an evaluator would need to predict which approach will provide the highest quality analysis of a sentence that needs to be processed in order to decide which one should be assigned the task. Because such a prediction would probably require at least a partial analysis of the sentence, we take the approach of assigning the task to all of the available language understanding approaches and then assessing the quality of the results relative to the expected typical accuracy faults of each approach.

The first competition model tries each approach in a preferred sequential ordering, stopping when a representation is acceptable according to a general filtering heuristic and otherwise continuing. The filtering heuristic estimates which representations are over or undergenerated and excludes those representations so that it appears that no representation was found for the sentence. A representation for a sentence is undergenerated if any of the word stems in a sentence are constants in the representation language and none of those are in the representation generated or if the representation produced is too sparse. For Why2-Atlas, it is too sparse if 50% of the propositions in the representation for a sentence have slots with less than two constants filling them. Most propositions in the representation language contain six slots which can be filled with constants. Propositions that are defined to have two or fewer slots that can be filled with constants are excluded from this assessment (e.g. the relations before and relposition are excluded). Representations are overgenerated if the sentences are shorter than 4 words since in general the physics principles to be recognized cannot be expressed in fewer words.

For the sequential model, we use a preference ordering of symbolic, statistical and hybrid in these experiments because of the way in which Why2-Atlas was originally designed and our expectations for which approach should produce the highest quality result at this point in the development of the knowledge sources. We also created some partial sequential models as well to look at whether the more expensive understanding approaches add anything significant at this point in their development.

The other competition model requests an analysis from all of the understanding approaches and then uses the filtering heuristic along with a ranking heuristic (as described below) to select the best analysis. If all of the analyses for either competition model fail to meet the selection heuristics then the sentence is regarded as uninterpretable. The run-time difference between the two competition models are nearly equivalent if each understanding approach in the second model is run in parallel using a distributed multi-agent architecture such as OAA [25].

The ranking heuristic again focus on the weaknesses of all the approaches. It computes a score for each representation by first finding the number of words in the intersection of the constants in the representation and the word stems in the sentence ($justified$), the number of word stems in the sentence that are constants in the representation language that do not appear in the representation ($undergenerated$) and the number of constants in the representation that are not word stems in the sentence ($overgenerated$). It then selects the one with the highest score, where the score is; $justified - 2 * undergenerated - .5 * overgenerated$. The weightings reflect both the importance and approximate nature of the terms.

The main difference between the two models is that the ranking approach will choose the better representation (as estimated by the heuristics) as opposed to one that merely suffices.

6 Results of the Combined Competing Approaches

The top part of Table 2 compares the baseline of tutoring all possible topics and the individual performances of the three understanding approaches when each is used in isolation from the others. We see that only the statistical approach lowers the false alarm rate but does so by sacrificing recall. The rest are not significantly different from tutoring all topics. However, the results of the statistical approach are clearly not good enough.

Table 2. Performance of individual language understanding approaches for actions taken in the Why2-Atlas system

approach	recall	precision	false alarm rate
baseline1 (tutor all topics)	1.0	.61	1.0
symbolic	1.0	.61	1.0
statistical (baseline2)	.60	.93	.07
hybrid	.94	.59	1.0
all (satisficing)	.67	.80	.26
hybrid + statistical (satisficing)	.70	.78	.31
symbolic + statistical (satisficing)	.69	.80	.26
all (highest ranked)	.73	.76	.36

The bottom part of Table 2, shows the results of combining the NL approaches. The satisficing model that includes all three NL mapping approaches performs better than the individual models in that it modestly improves recall but at the sacrifice of a higher false alarm rate. The satisficing model checks each representation in order 1) symbolic 2) statistical 3) hybrid, and stops with the first representation that is acceptable according to the filtering heuristic. We also see that both of the satisficing models that include just two understanding approaches perform better than the model in which all approaches are combined; with the symbolic + statistical model being the best since it increases recall without further increasing the false alarm rate. Finally, we see that the model, which selects the best representation from all three approaches, provides the most balanced results of the combined or individual approaches. It provides the largest increase in recall and the false alarm rate is still modest compared to the baseline of tutoring all possible topics. To make a final selection of which combined approach one should use, there needs to be an estimate of which errors will have a larger negative impact on student learning. But clearly, selecting a combined approach will be better than selecting a single NL mapping approach.

7 Discussion and Future Work

Although none of the NL mapping approaches adequately represent the physics content covered by the Why2-Atlas system at this point in their knowledge de-

velopment, they can be combined advantageously by estimating representations that are over or undergenerated.

We are considering two future improvements. One is to automatically learn ranking and filtering heuristics using features that represent differences between annotated representations and the representations produced by the understanding approaches. The heuristics can then be tuned to the types of representations that the approaches are producing at a particular time-slice in the domain knowledge development. The second future improvement is to add reference resolution to the heuristics in order to canonicalize words and phrases to their body constants in the representation language. Although we could try canonicalizing other lexical items to their representation language constants, this might not be as fruitful. While a physics expert could use *push* and *pull* and know that this implies that forces are involved, this is not a safe assumption for introductory physics students.

8 Acknowledgments

This research was supported by ONR Grant No. N00014-00-1-0600 and by NSF Grant No. 9720359.

References

1. VanLehn, K., Jordan, P., Rosé, C., Bhembe, D., Böttner, M., Gaydos, A., Makatchev, M., Pappuswamy, U., Ringenberg, M., Roque, A., Siler, S., Srivastava, R.: The architecture of Why2-Atlas: A coach for qualitative physics essay writing. In: Proceedings of Intelligent Tutoring Systems Conference. Volume 2363 of LNCS., Springer (2002) 158–167
2. Aleven, V., Popescu, O., Koedinger, K.: Pilot-testing a tutorial dialogue system that supports self-explanation. In: Proceedings of Intelligent Tutoring Systems Conference. Volume 2363 of LNCS., Springer (2002) 344
3. Zinn, C., Moore, J.D., Core, M.G.: A 3-tier planning architecture for managing tutorial dialogue. In: Proceedings of Intelligent Tutoring Systems Conference (ITS 2002). (2002) 574–584
4. Makatchev, M., Jordan, P., VanLehn, K.: Abductive theorem proving for analyzing student explanations and guiding feedback in intelligent tutoring systems. Journal of Automated Reasoning: Special Issue on Automated Reasoning and Theorem Proving in Education (2004) to appear.
5. Aleven, V., Popescu, O., Koedinger, K.R.: A tutorial dialogue system with knowledge-based understanding and classification of student explanations. In: Working Notes of 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems. (2001)
6. Sandholm, T.W.: Distributed rational decision making. In Weiss, G., ed.: Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. The MIT Press, Cambridge, MA, USA (1999) 201–258
7. Ploetzner, R., VanLehn, K.: The acquisition of qualitative physics knowledge during textbook-based physics training. Cognition and Instruction **15** (1997) 169–205

8. Walther, C.: A many-sorted calculus based on resolution and paramodulation. Morgan Kaufmann, Los Altos, California (1987)
9. Rosé, C.P.: A framework for robust semantic interpretation. In: Proceedings of the First Meeting of the North American Chapter of the Association for Computational Linguistics. (2000) 311–318
10. McCallum, A., Nigam, K.: A comparison of event models for naive bayes text classification. In: Proceeding of AAAI/ICML-98 Workshop on Learning for Text Categorization, AAAI Press (1998)
11. Jordan, P.W.: A machine learning approach for mapping natural language to a domain representation language. in preparation (2004)
12. Abney, S.: Partial parsing via finite-state cascades. Journal of Natural Language Engineering **2** (1996) 337–344
13. Lin, D.: Dependency-based evaluation of MINIPAR. In: Workshop on the Evaluation of Parsing Systems, Granada, Spain (1998)
14. Levin, B., Pinker, S., eds.: Lexical and Conceptual Semantics. Blackwell Publishers, Oxford (1992)
15. Jackendoff, R.: Semantics and Cognition. Current Studies in Linguistics Series. The MIT Press (1983)
16. Rosé, C., Gaydos, A., Hall, B., Roque, A., VanLehn, K.: Overcoming the knowledge engineering bottleneck for understanding student language input. In: Proceedings of of AI in Education 2003 Conference. (2003)
17. Dzikovska, M., Swift, M., Allen, J.: Customizing meaning: building domain-specific semantic representations from a generic lexicon. In Bunt, H., Muskens, R., eds.: Computing Meaning. Volume 3. Academic Publishers (2004)
18. Rosé, C., Roque, A., Bhembe, D., VanLehn, K.: A hybrid text classification approach for analysis of student essays. In: Proceedings of HLT/NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing. (2003)
19. Lin, D., Pantel, P.: Discovery of inference rules for question answering. Journal of Natural Language Engineering **Fall-Winter** (2001)
20. Strube, M., Rapp, S., Müller, C.: The influence of minimum edit distance on reference resolution. In: Proceedings of Empirical Methods in Natural Language Processing Conference. (2002)
21. Ng, V., Cardie, C.: Improving machine learning approaches to coreference resolution. In: Proceedings of Association for Computational Linguistics 2002. (2002)
22. Flach, P.: The geometry of ROC space: Understanding machine learning metrics through ROC isometrics. In: Proceedings of 20th International Conference on Machine Learning. (2003)
23. MacMillan, N., Creelman, C.: Detection Theory: A User's Guide. Cambridge University Press, Cambridge, UK (1991)
24. Franklin, S., Graesser, A.: Is it an agent, or just a program?: A taxonomy for autonomous agents. In: Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag (1996)
25. Cheyer, A., Martin, D.: The open agent architecture. Journal of Autonomous Agents and Multi-Agent Systems **4** (2001) 143–148
26. Jokinen, K., Kerminen, A., Kaipainen, M., Jauhiainen, T., Wilcock, G., Turunen, M., Hakulinen, J., Kuusisto, J., Lagus, K.: Adaptive dialogue systems - interaction with interact. In: Proceedings of the 3rd SIGdial Workshop on Discourse and Dialogue. (2002)