LANGUAGE PROCESSING CHALLENGES AND SOLUTIONS IN INTELLIGENT TUTORING SYSTEMS WITH NATURAL LANGUAGE INTERACTION

Vasile RUS

University of Memphis
Department of Computer Science
E-mail: vrus@memphis.edu

Abstract. We provide an overview of Intelligent Tutoring Systems that interact with students through natural language dialogue. We present the major challenges such systems face when it comes to processing natural language student input and present an overview of existing solutions to these challenges. In particular, we emphasize an approach, lexico-syntactic entailment (LSE), to sentence-level processing in tutoring systems. Additionally, we illustrate the impact of natural language components on other tutoring systems modules such as student model, feedback, and domain model.

Key words: natural language processing, intelligent tutoring systems.

1. INTELLIGENT TUTORING SYSTEMS WITH NATURAL LANGUAGE INTERACTION

1.1. Introduction

Complex learning environments such as intelligent tutoring systems (ITSs) depend on natural language understanding and assessment for fast and accurate interpretation of human language so that the system can respond intelligently in natural language. Such ITSs function by interpreting the meaning of student input, assessing the extent to which it manifests learning, and generating suitable feedback to the learner. To operate effectively, the accuracy of assessing student input is critical because inaccurate feedback can potentially compromise learning and lower the student's motivation and metacognitive awareness of the learning goals of the system (Millis *et al*., 2007). At the same time, systems need to be fast enough to operate in the real time environments of ITSs. Delays in interactivity

caused by computational processing run the risk of frustrating the user and leading to lower engagement with the system. The natural language approach for interpreting user input must also be scalable. That is an ITS needs to be able to operate with large and diverse data sets and to be ported from one subject/domain to another. Thus, a scalable ITS can reach out to more students from more disciplines and easily incorporate new material as the learning environment develops. As such, student input in ITSs requires an assessment approach that is accurate enough to provide appropriate evaluation, fast enough to operate in real time, and flexible enough to be easily ported. An approach we proposed, lexico-syntactic entailment (LSE; Rus & Graesser, 2006; Rus *et al*., 2008; 2009), meets these requirements.

1.2. What is an Intelligent Tutoring System?

Intelligent Tutoring Ssytems (ITSs) can be distinguished from more conventional computer based training (CBT) systems by the intelligence of the computational mechanisms that track the student's performance and adaptively respond. Both ITSs and most CBT systems have a number of features that are designed to promote learning: (a) collect a profile of the student's performance, skills, knowledge, mastery of specific content, and general student attributes, (b) give feedback on performance and errors, and (c) select actions, problems, and tasks that are sensitive to the student's profile. The mechanisms in CBT systems are comparatively simple: there are a small number of states and parameters in the student profile, a small number of feedback options, and simple algorithms for action selection (such as conditional branching in a small space of action options). ITSs mechanisms are more complex in the grain size of profile states, the formulation of feedback, and action selection algorithms. The process of *tracking knowledge* (called *user modeling*) and the process of *adaptively responding to the learner* ideally incorporate advanced computational models in artificial intelligence and cognitive science, such as *production systems*, *Bayes networks*, *structured-* or *statistical- representations of knowledge*, *theorem proving*, *case-based reasoning*, *induction*, *inference*, and *constraint satisfaction algorithms*. Meta-analyses have revealed that these ITSs fare well when effect sizes are measured that compare the computer systems to classroom instruction and other suitable controls (Dodds & Fletcher, 2004; Fletcher, 2003; Wisher & Fletcher, 2004): the effect sizes are 0.39 for CBT, 0.50 for multimedia, and 1.08 for ITSs. An effect size of 1.00 means one letter-grade improvement, *e.g*. from B to A or from C to B.

The added intelligence of ITSs has produced some notable successes. Successful systems have been developed for mathematically well-formed topics, including algebra, geometry, programming languages (the Cognitive Tutors, Anderson *et al*., 1995; Koedinger *et al*., 1997), physics (Andes, Atlas, and Why/Atlas, VanLehn *et al*., 2005; 2007), electronics (Lesgold *et al*., 1992), and

information technology (Mitrovic *et al*., 2004). These systems show impressive learning gains (1.00 sigma, approximately, *i.e*. one letter-grade improvement), particularly for deeper levels of comprehension. School systems are adopting ITSs at an increasing pace, particularly those developed at LearnLab (an NSF-funded Science of Learning Center) and Carnegie Learning. These centers are scaling up their ITSs in mathematics, physics, and foreign languages. Some of the recent ITSs have attempted to handle knowledge domains that are not mathematically precise and well-formed. The Intelligent Essay Assessor (Foltz *et al*., 2000; Landauer, 2007) and e-Rater (Burstein, 2003) grade essays on science, history, and other topics as reliably as experts of English composition. Summary Street (Kintsch *et al*., 2007) helps the learner summarize texts by identifying idea gaps and irrelevant information. AutoTutor (Graesser *et al*., 2004; 2005) helps college students learn about computer literacy, physics, and critical thinking skills by holding conversations in natural language. iSTART (McNamara *et al*., 2004) trains students on reading strategies using science, *e.g*. biology, texts.

1.3. How do ITSs Understand and Assess Natural Language Input?

One of the ways in which ITSs with natural language understanding verify student input is through *matching*. In some cases, the match is between the user input and a pre-selected *stored answer to a question*, *solution to a problem*, *misconception*, or other form of *benchmark response*. In other cases, the system evaluates the degree to which the student input varies from a complex representation or a dynamically computed structure. The computation of matches and similarity metrics are limited by the fidelity and flexibility of the natural language processing modules.

The major challenge with assessing natural language input is that it is relatively unconstrained and rarely follows brittle rules in its computation of spelling, syntax, and semantics. Researchers who have developed tutorial dialogue systems in natural language have explored the accuracy of matching students' written input to targeted knowledge. Examples of these systems are AutoTutor and Why-Atlas, which tutor students on Newtonian physics as already mentioned (Graesser *et al*., 2005; VanLehn *et al*., 2007), and the iSTART system, which helps students read text at deeper levels – also mentioned earlier (McNamara *et al*., 2004). Systems such as these have typically relied on statistical representations, such as latent semantic analysis (LSA; Landauer *et al*., 2007) and content word overlap metrics (McNamara *et al*., 2007). Indeed, such statistical and word overlap algorithms can boast much success. However, over short dialogue exchanges (such as those in ITSs), the accuracy of interpretation can be seriously compromised without a deeper level of lexico-syntactic textual assessment (McCarthy *et al*., 2007). Such a lexico-syntactic approach, *entailment evaluation* (Rus & Graesser, 2006; Rus *et al*., 2008; 2009), is our proposal to meet the challenge of natural

language understanding and assessment in intelligent tutoring systems. Our approach to lexico-syntactic entailment (LSE) incorporates deeper natural language processing solutions for ITSs with natural language exchanges while remaining both sufficiently fast to provide real time assessment of user input, and sufficiently flexible to be ported.

How does our approach differ to existing approaches? Current evaluation approaches tend to fall into one of two categories: *logic-based* and *lexical-based*. Logic-based approaches for interpreting user input in ITSs, such as the abduction-based approach used in the Why/Atlas tutoring system (VanLehn *et al*., 2002), are accurate but less interactive and less scalable. Similarly, lexical-based approaches, such as Latent Semantic Analysis (LSA; Landauer *et al*., 2007) or word overlap used in two state-of-the-art ITSs, AutoTutor (Graesser *et al*., 2005) and iSTART (McNamara *et al*., 2007), are scalable and interactive but less precise at interpreting natural language input. Thus, an ideal system would have an approach as accurate as logic-based approaches and as scalable as lexical-based approaches.

We proposed an approach, lexico-syntactic entailment (LSE; Rus & Graesser, 2006; Rus *et al*., 2008; 2009), that is scalable and provides accurate and real-time interpretation of natural language user input in ITSs. The approach strikes a balance between accuracy of logic-based approaches on one hand, and scalability and interactivity of lexical-based approaches, on the other hand. A unique aspect of our proposed approach is the weighted combination of lexical information (*lexical component*) with symbolic representations of syntax (*syntactic component*), including negation handling (*negation*). The proposed approach can handle explicit and implicit negation. Negation handling is an important and novel feature of our approach. Negation handling is not detailed here. We will investigate the accuracy of detecting negation and the impact of the negation handling component on the overall accuracy of the natural language understanding approach in the future. In addition to accuracy, interactivity, and scalability, the proposed approach is also *customizable* by tuning the weights of the different components of the LSE approach. It is important to have a customizable approach so that it can be adapted to various goals. For instance, Rus and Graesser (2006) showed that a word overlap method enhanced with synonymy provided the highest confidence results (the LSE approach provided best accuracy) on evaluating student answers in AutoTutor, while syntax leads to highly-precise outcomes. If a highly-confident system is needed for the task of evaluating student answers in AutoTutor then full weight should be given to the lexical component and zero weight to the syntactic component. Furthermore, some components can be turned on and off. Lastly, as we just mentioned, the LSE approach proposed here offers a measure of confidence in its assessment decisions. These confidence measures will be used to provide feedback to learners. For instance, two student inputs which are assessed as close to the ideal answer will lead to different feedback depending on the confidence on that assessment. If there is high confidence that one input is really close to the ideal

answer then positive feedback should be offered. If the confidence in the closeness decision is low then we should probably give neutral feedback or try to make the student rephrase his answer.

2. PREVIOUS WORK ON INTERPRETING NATURAL LANGUAGE INPUT IN ITSs

ITSs with natural language interaction use various techniques for natural language understanding and assessment. The techniques can be grouped in two broad categories: *lexical-based* and *logic-based*. The first category includes word overlap, Latent Semantic Analysis (LSA), and other, similar techniques. The second category (logic-based) comprises logical-proof-based methods that rely on deep, symbolic representation of language, *e.g*. the abduction-based methods.

2.1. Lexical-based Approaches

Lexical-based approaches to assessment in ITSs (*e.g*. word overlap, LSA, Kendall's tau (Dennis, 2007)) use only words as their primitives of understanding. Such approaches can be described as *basic lexical-based* approaches. *Hybrid* or *combined lexical-based* approaches include two or more of the basic approaches. As an example of a hybrid lexical based-approach, McNamara *et al*. (2007) combined LSA with word overlap in a weighted-sum to evaluate student input, called *self-explanations*, in iSTART.

Word overlap indices compute a score that reflects the fraction of common words, or content words, that co-occur across two text fragments. For instance, the denominator of the fraction could be the average number of words in the two fragments. The advantage of this approach is that it is scalable because it can be applied to sentence level or paragraph level assessment and because it can be easily ported from one domain to another. The method is also simple and robust, and it has been used relatively successfully for some assessment tasks in ITSs. As with iSTART, McNamara *et al*. (2007) showed that word-overlap performed relatively well at identifying poor self-explanations and paraphrases, a particular type of self-explanation in iSTART. In addition, Graesser *et al*. (2007) found that a simple word overlap metric does a somewhat better job than LSA when sentential units are assessed. However, despite their relative success, word overlap methods are not highly accurate because they do not address issues such as anaphora or word order. If a self-explanation in iSTART does not contain exact words from the text base, although it may be semantically close to it, word overlap performs poorly.

Latent Semantic Analysis (LSA) is a statistical technique for text understanding that relies on word co-occurrences to detect related words. It is based on the principle that the meaning of a word is defined by the company it

keeps. Two words have related meaning if they co-occur in the same contexts. The co-occurrence information is derived from large collections of text documents. Each text fragment could be represented as a vector in a high-dimensional space. There is one dimension for each unique word in the collection. Typically, LSA uses singular value decomposition (Landauer *et al*., 2007) to reduce the dimensionality of the space to 300-500 dimensions. The semantic relatedness of two text fragments is computed as the cosine between the corresponding LSA vectors. Similar to word-overlap methods, LSA is a promising technique for assessing multi-sentence student contributions or to port a system from one domain to another. It has the advantage of being trained on nothing more than natural language text from a particular domain. LSA has had remarkable success in capturing the world knowledge that is needed for grading essays of students (Foltz, 1996) and in matching texts to students of varying abilities to optimize learning (Wolfe *et al*., 1998). Kintsch's construction-integration model of text comprehension has incorporated LSA as a major component in its knowledge construction phases (Kintsch, 1998). In ITSs, LSA has been successfully used for assessing and improving reading comprehension (Millis *et al*., 2007), to evaluate self-explanations in iSTART (McNamara et al., 2007), and to evaluating answer essays in AutoTutor (Graesser *et al*., 2007). LSA-based similarity metrics are capable of evaluating the quality of learner contributions almost as well as graduate students in the subject matter, i.e., physics or computer literacy (Graesser *et al*., 2004). McNamara *et al*. (2007) reported that LSA leads to a more stable performance than word-overlap methods across different types of texts.

Hybrid lexical-based approaches, that is LSA in combination with other lexical-based methods, such as word overlap, provides better results over LSA or word overlap alone. iSTART experiments (McNamara *et al*., 2007) have shown that LSA combined with word overlap was the most successful method for evaluating and categorizing self-explanations with respect to human expert ratings. Graesser *et al.* (2007) found that a combined approach using LSA, word-overlap, and Kendall's Tau leads to better correlations with expert judges' ratings on matching sentential ideal answers to learner's sentential contributions in AutoTutor.

While lexical-based approaches have had notable success, they cannot account for a number of language phenomena, *e.g*. anaphora, or distant syntactic relationship, called *remote dependencies*, among words in a sentence. Therefore, it is important to have student input represented in a way that can be interpreted at a deep level. Our proposed approach adds symbolic representations of syntax and negation handling on top of a lexical-based approach, *i.e*. world overlap, for a deeper, more precise representation of natural language input in ITSs. A deeper representation leads to more accurate interpretations of user input and thus better feedback resulting in better learning.

Specific Problems with Lexical-Based Approaches. We describe below five specific problems with assessing natural language input based on lexical

approaches. We focus on lexical approaches because they are the current approaches used in the ITSs we focus on in this chapter: AutoTutor and iSTART.

Text length. Text length is a widely acknowledged confound that needs to be accommodated by all text measuring indices. The performance of syntactic parsers critically depends on text length (Jurafsky & Martin, 2000). This in turn affects feedback accuracy and interactivity of ITSs. As another example, lexical diversity indices (such as type-token ratio) are sensitive to text length because as the length of text increases the likelihood of new words being incorporated into the text decreases (McCarthy & Jarvis, 2007; Tweedie & Baayen, 1998). This length problem is similar for text relatedness measures such as LSA and overlap-indices: Given longer texts to compare, there is a greater chance that similarities will be found (Dennis, 2007; McNamara *et al*., 2006; Penumatsa *et al*., 2004; Rehder & Hastie, 1998). As a consequence, the analysis of short texts, such as those created in ITS environments, appears to be particularly problematic (Wiemer-Hastings *et al*., 1999). The upshot of this problem is that longer responses tend to be judged by the ITSs as closer to an ideal set of answers retained within the system. Consequently, a long (but wrong) response can receive more favorable feedback than one that is short (but correct). Our LSE approach is based on a normalized metric that uses the length of one of the text fragments as the normalization factor. This makes the LSE approach less susceptible to the text length confound.

Typing errors. It is unreasonable to assume that students using ITSs should have perfect writing ability. Indeed, student input has a high incidence of misspellings, typos, grammatical errors, and questionable syntactical choices. Current relatedness indices do not cater to such eventualities and assess a misspelled word as a very rare word that is substantially different from its correct form. When this occurs, relatedness scores are adversely affected, leading to negative feedback based on spelling rather than understanding of key concepts. A simple solution to spelling, for instance, is to use an automatic spell checker. For grammatical errors, there is no immediate solution.

Negation. For measures such as LSA and content word overlap, the sentence *the man is a doctor* is considered very similar to the sentence *the man is not a doctor*, although semantically the sentences are quite different. Antonyms and other forms of negations are similarly affected. In ITSs, such distinctions are critical because inaccurate feedback to students can seriously effect motivation (Graesser & Person, 1994). The LSE approach can handle explicit and implicit negation forms in natural language.

Syntax. For both LSA and overlap indices, *the dog chased the man* and *the man chased the dog* are viewed as identical. ITSs are often employed to teach the relationships between ideas (such as causes and effects), so accurately assessing syntax is a high priority for computing effective feedback. In our LSE approach, syntax is captured explicitly in the form of dependencies among the words.

Asymmetrical issues. Asymmetrical relatedness refers to situations where sparsely-featured objects are judged as less similar to general- or multi-featured objects than *vice versa*. For instance, *poodle* may indicate *dog* or *Korea* may signal *China* while the reverse is less likely to occur (Tversky, 1977). The issue is important to text relatedness measures, which tend to evaluate lexico-semantic relatedness as being equal in terms of reflexivity. Intelligent Tutoring Systems need to understand such differences and distinguish the direction of relationships. Thus, accurate feedback can be given to students depending on whether they are generalizing a rule from specific points (summarizing) or making a specific point from a general rule (elaborating).

We proposed a novel approach, LSE (Rus & Graesser, 2006; Rus *et al.*, 2008, 2009), that addresses the above problems in assessment tasks in ITSs. The approach will be presented in detail in section 3.

2.2. Logical-based Approaches

Previous work on deeper processing in ITSs (*e.g.*, Why-Atlas; VanLehn *et al*., 2002; Jordan & VanLehn, 2002; Makatchev *et al*., 2004) has focused on modeling students' reasoning about qualitative physics as abductive proofs. The Why-Atlas tutoring system (VanLehn *et al*., 2002) uses Tacitus-lite+, a variant of the theorem prover Tacitus-lite (Hobbs *et al*.,, 1988), for abductive reasoning. Abduction is inference from an effect or observation to possible causes or explanations. In the Why-Atlas system, the observations are what the student says and the possible explanations are the background knowledge, *i.e*. physics laws, and line of reasoning that would support what the student said. Some assumptions must be made during abduction to link the explanations to observations. From the set of possible explanations, the least expensive explanation is chosen.

The abduction-based method in Why–Atlas addresses the need for deeper natural language understanding in ITSs. Jordan and VanLehn (2002) remarked that LSA-based approaches "are not yet able to distinguish subtle but important differences between good and bad explanations". They went on to write "Statistical classification is insensitive to negations, anaphoric references, and argument ordering variations and its inferencing is weak". Symbolic approaches could offer the necessary precision to capture subtleties of language. The Why–Atlas system is based on a symbolic approach that interprets language at a deeper level to capture fine aspects of language expressions. The abduction-based method in Why–Atlas relies on a sentence-level understanding component (Rosé, 2000; Freedman *et al*., 2000) and a discourse-level understanding module that uses language and domain reasoning axioms. The discourse-level understanding module relies on the language axioms and abductive inference to transform sentence-level propositions onto discourse-level propositions that are more complete, for instance anaphors being resolved in such discourse-level propositions.

While providing precise interpretations of user input, there are several problems with the abduction-based approach in Why–Atlas: portability, scalability, and interactivity. Jordan and VanLehn (2002) indicate that Why–Atlas had 90 language axioms and 95 domain axioms. The domain axioms fully cover 5 problems they experimented with. Handling new problems and new domains would require extra manually-entered domain axioms, which makes portability of the system expensive. Scaling the approach to large sets of problems would require a large number of axioms to be manually developed, leading to high costs and long periods of axiom-development time. The solution to make the approach more scalable and portable is to have automated procedures for generating the axioms and representations of user input. Jordan and VanLehn (2002) automatically derive the deep representations for natural language user input but do not report how accurately the procedure is or how expensive it is to move from one set of physics problems to ten more sets or from one subject to another. Obviously, evaluating the accuracy of a complex representation is a problem of itself. Currently, there is no reported automated method to generate axioms in ITSs. Rus (2001; 2002) does present a general method to derive world axioms from WordNet but it is not clear how the method would transfer to specific domains, *e.g*. physics, in ITSs. In terms of interactivity, the average waiting time in the best case scenario (Jordan & VanLehn, 2002) for a student to get feedback from Why–Atlas is 21.22 seconds after completing the last sentence in the essay. Such relatively long waiting time could frustrate the user leading to less interest and thus less impact on learning. A larger set of axioms would eventually lead to even longer delays, which would make the waiting time even longer.

In sum, there is a trade-off between the two lexical and logical categories of approaches. Lexical-based methods are scalable but less accurate. They are suitable for assessing multi-sentence student explanations. The use of lexical-based methods makes ITSs robust and easily transferable/portable from one problem to another, from one domain to another. Nonetheless, these approaches treat language as a independent words so they fail to account for a number of language phenomena such as anaphora and long-distance (remote) dependencies. The logical-proof-based methods, on the other hand, are deep and can provide an explanation for why the student answer is deemed correct (or not). The explanation is automatically derived using automated *provers* that use symbolic representations of language and domain knowledge in the form of axioms. The problem with logical-proof-based methods is that they are not scalable and require manual intervention in some processing steps, *e.g*. to develop the discourse and domain axioms in Why–Atlas (Jordan & VanLehn, 2002). One other problem with existing logical-proof-based methods is that they are less appropriate for interactive situations, where a fast response is required. We proposed a representation that balances depth of language understanding in logical-proof-based methods with the interactivity, scalability, and portability of lexical-based methods. The goal is to

improve the accuracy of lexical-based approaches, by employing syntax and negation processing modules, while preserving the scalability and interactivity features of these approaches. Ideally, we would end up with an approach as accurate as logic-based approaches and as scalable as lexical-based approaches.

2.3. Human Performance to Assessing Students' Contributions

It is known that automated approaches to natural-language interpretation are not perfect. Because the feedback provided to learners in ITSs relies on the accuracy of interpreting natural language user input there is the legitimate concern that imperfect interpretation would lead to improper feedback with negative effects on learning.

This worry has proven not to be entirely justified by current automated approaches to assessment in ITSs based on LSA, word-overlap, or hybrid. Such approaches have led to learning gains (Graesser *et al*., 2007; McNamara *et al*., 2007; Millis *et al*., 2007) although their accuracy in interpreting natural language input is far from perfect. Our proposed approach improves the accuracy of interpreting user input and thus potentially leading to more accurate feedback and greater learning gains.

It is important, when addressing the above concern, to put automated approaches to interpreting natural language user input in ITSs into perspective. Benchmarks of human language understanding are the humans. We thus have to look at how good humans are at interpreting user input in ITSs. There is evidence that correlations between an ITS and a human for interpreting user input in ITSs is close to human-human correlations, even though the ITS' interpretation of student input is not perfect. Prior analyses have shown that AutoTutor's LSA component performs conceptual pattern matching operations almost as well as human judges (Graesser *et al*., 2005). When AutoTutor grades the percentage of expectations (sentences that together form an ideal answer to a problem) that are covered in an answer to a question/problem, the correlation between AutoTutor and a graduate student expert has been approximately r = .50, whereas two graduate students correlate at r = .60. In a related experiment (Graesser *et al*., 2007), correlation between a pair of experts was approximately r = .65 when making judgments about the quality of student essays in AutoTutor. Correlations between LSA and judges' quality ratings were approximately r = .50. Millis *et al*. (2007) report inter-rater reliability of r = .80 on a task of categorizing self-explanations.

3. LEXICO-SYNTACTIC ENTAILMENT

The lexico-syntactic entailment (LSE) approach is a fully automated algorithm that combines lexical, syntactic, and negation information in a single measure of text relatedness between two sentences. The measure is asymmetrical and although it was initially developed for the task of recognizing textual

entailment (Dagan *et al*., 2005), it has been adapted for various text-to-text relatedness tasks, such as entailment-, paraphrase- and elaboration- evaluation.

Entailment evaluations help in the assessment of the appropriateness of student responses during ITS exchanges. Entailment can be distinguished from three similar terms (implicature, paraphrase, and elaboration), all of which are also important for assessment in ITS environments. The terms *entailment* is often associated with the highly similar concept of *implicature*. The distinction is that entailment is reserved for linguistic-based inferences that are closely tied to *explicit* words, syntactic constructions, and formal semantics, as opposed to the knowledge-based *implied* referents and references, for which the term implicature is more appropriate. Implicature corresponds to the controlled knowledge-based elaborative inferences defined by Kintsch (1993) or to knowledge-based inferences defined in the inference taxonomies in discourse psychology (Graesser & Person, 1994).

The term *paraphrase* and *elaboration* also need to be distinguished from entailment. A *paraphrase* is a reasonable restatement of the text. Thus, a paraphrase is a form of entailment, yet an entailment is not necessarily a paraphrase. This asymmetric relation can be understood if we consider that *John went to the store* is entailed by (but not a paraphrase of) *John drove to the store to buy supplies*. The term *elaboration* refers to information that is generated inferentially or associatively in response to the text being analyzed, but without the systematic and sometimes formal constraints of entailment, implicature, or paraphrase. Examples of each term are provided below for the sentence

John drove to the store to buy supplies.

Entailment: *John went to the store.*
(Explicit, logical implication based on the text)

Implicature: *John bought some supplies*.
(Implicit, reasonable assumption from the text, although not explicitly stated in the text)

Paraphrase: *He took his car to the store to get things that he wanted.*
(Reasonable re-statement of all and only the critical information in the text)

Elaboration: *He could have borrowed stuff.*
(Reasonable *reaction* to the text)

Evaluating entailment is generally referred to as the task of *recognizing textual entailment* (RTE; Dagan *et al*., 2005). Specifically, it is the task of deciding, given two text fragments, whether the meaning of one text logically infers the other. When it does, the evaluation is deemed as T (the entailing text) entails H (the

entailed hypothesis). For example, a text (from the RTE data) of *Eyeing the huge market potential, currently led by Google, Yahoo took over search company Overture Services Inc last year* would entail a hypothesis of *Yahoo bought Overture*. The task of recognizing entailment is relevant to a large number of applications, including machine translation, question answering, and information retrieval.

The task of textual entailment has been a priority in investigations of information retrieval (Monz & de Rijke, 2001) and automated language processing (Pazienza *et al*., 2005). In related work, Moldovan and Rus (2001) analyzed how to use unification and matching to address the *answer correctness* problem. Similar to entailment, *answer correctness* is the task of deciding whether candidate answers logically imply an ideal answer to a question.

A complete solution to the textual entailment challenge requires linguistic information, reasoning, and world knowledge (Rus *et al*., 2009). Our approach focuses on the role of linguistic information in making entailment decisions. The overall goal is to produce a light (*i.e*. computationally inexpensive), but accurate solution that could be used in interactive systems such as ITSs. Solutions that rely on processing-intensive deep representations (*e.g*., frame semantics and reasoning) and large structured repositories of information (*e.g*., ResearchCyc) are impractical for interactive tasks because they result in lengthy response times, causing user dissatisfaction.

Our solution for recognizing textual entailment is based on subsumption: In general, an object X subsumes an object Y if and only if X is more general than or identical to Y. Applied to textual entailment, subsumption translates as follows: hypothesis H is entailed from T if and only if T subsumes H.

To illustrate the main idea of our approach, we pick the Text-Hypothesis pair below (adapted from an example in RTE (Dagan *et al*., 2005)).

T: *The bombers managed to enter the embassy compounds.*
H: *The bombers entered the embassy compounds.*

In this case, the Text T entails H because T contains every word in H and all relationships among these words. For instance, the *subject* relationship between *bombers* and *enter* in H is present in T. We say H is contained in T and, as a consequence, knowing T means that we also know H. The fewer the words and relationships in H that are contained in T, the less the chance the meaning of H being contained/implied by T.

The subsumption-based solution to entailment has two phases: (I) map both T and H into graph structures and (II) perform a subsumption operation between the T-graph and H-graph. An entailment score, entail(T,H), is computed, quantifying the degree to which the T-graph subsumes the H-graph.

In phase I, the two text fragments involved in a textual entailment decision are initially mapped onto a graph representation. The graph representation employed is based on the dependency-graph formalisms of Mel'cuk (1998). The

mapping relies on information from syntactic parsers. Syntactic parsing in its most general definition may be viewed as discovering the underlying syntactic structure of a sentence. The specificities include the types of elements and relations that are retrieved by the parsing process and the way in which they are represented. For example, Treebank-style (Marcus *et al*., 1993) parsers retrieve a bracketed form that encodes a hierarchical organization (tree) of smaller elements (called phrases; hence these parsers are called phrase-based), while Grammatical-Relations (GR)-style parsers explicitly output relations together with elements involved in the relation (subj(John,walk)). The output of phrase-based parsers can be mapped into a set of explicit relations by linking the head of each phrase to its modifiers in a systematic mapping process. A *dependency tree* is thus generated. The dependency tree encodes exclusively local dependencies (head-modifiers), as opposed to long-distance (*remote*) dependencies, such as the remote subject relation between *bombers* and *enter* in the sentence *The bombers managed to enter the embassy compounds*. Thus, in this stage, the dependency tree is transformed onto a *dependency graph* by generating *remote dependencies* between content words. Remote dependencies are computed by a naïve-Bayes functional tagger (Rus & Desai, 2005). An example of a dependency graph is shown in Figure 1 for the sentence *The two objects will cover the same horizontal distance*. For instance, there is a subject (*subj)* dependency relation between *objects* and *cover*. If a dependency parser that can generate remote dependencies is used, instead of a phrase-based parser, there is no need to explicitly derive the remote dependencies using a functional tagger.

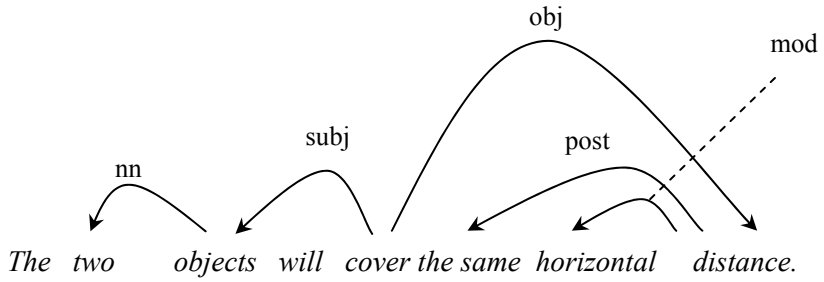

Figure 1. An example of a dependency graph. (subj – subject;
nn – complex nominal; obj – object; post – postmodifier; mod - modifier).

In *phase II*, the textual entailment problem (*i.e*., each T and H) is mapped into a specific example of graph isomorphism called *subsumption* (also known as *containment*). Isomorphism in graph theory addresses the problem of testing whether two graphs are the same.

A graph G = (V, E) consists of a set of nodes or *vertices* V and a set of *edges* E. Graphs can be used to model the linguistic information embedded in a sentence:

vertices represent concepts (*e.g*., *bombers*, *joint venture*) and edges represent syntactic relations among concepts (*e.g*., the edge labeled *subj* connects the verb *cover* to its subject *objects*). The Text (T) entails the Hypothesis (H) if and only if the hypothesis graph is subsumed (or contained) by the text graph.

The subsumption algorithm for textual entailment (Rus *et al*., in press) has three major steps: (1) find an isomorphism between *V*H (set of vertices of the Hypothesis graph) and *V*T; (2) check whether the labeled edges in H, *E*H, have correspondents in *E*T ; and (3) compute score. In step 1, for each vertex *V*H, a correspondent *V*T node is sought. If a vertex in H does not have a direct correspondent in T, a thesaurus is used to find all possible synonyms for vertices. Step 2 takes each relation in H and checks its presence in T. The checking is augmented with relation equivalences among linguistic phenomena such as possessives and linking verbs (*e.g*. *be, have*). For instance, *tall man* would be equivalent to *man is tall*. A normalized score for vertices and edge mapping is then computed. The score for the entire entailment is the sum of each individual vertex and edge matching score. Finally, the score must account for negation. The approach handles both *explicit* and *implicit* negation. Explicit negation is indicated by particles such as *no, not, neither ... nor* and the shortened form *n't*. Implicit negation is present in text via deeper lexico-semantic relations among linguistic expressions. The most obvious example is the *antonymy* relation among words, which is retrieved from WordNet (Miller, 1995). Negation is accommodated in the score after making the entailment decision for the Text-Hypothesis pair (without negation). If any one of the text fragments is negated, the decision is reversed, but if both are negated the decision is retained (double-negation), and so forth. We acknowledge the fact that other forms of negation exists. For instance, knowing that an object is *not* hot does not entail that the object is cold (*i.e*., it could simply be warm). We plan to extend the current negation handling module to address other forms of negation.

The Scoring. The formula for an overall entailment value aims to quantify the degree of entailment between T and H. Scores above a certain threshold are deemed as TRUE entailment and scores below are deemed FALSE entailment. Based on extensive testing, Rus, McCarthy, McNamara, & Graesser (in press) found that .50 is the best threshold for RTE-like tasks. The higher the score above .50, the greater the *confidence* of the entailment decision. Similarly, the lower the score below .50, the greater the confidence in the FALSE decision (see in Rus *et al*., 2007). The three components of the score are lexical or node matching, syntactic or relational matching, and negation. The weights of lexical and syntactic matching are given by parameters α and β, respectively (see Equation 1). Another parameter in the score is the free term γ which can be used to bias scores.

The effect of negation on entailment decision is captured by the last term of the equation. The choice of α, β and γ can have a large impact on the overall score. From its definition, the entailment score is non-symmetrical, entscore(H, T) ≠

entscore(T,H), because it is normalized on the basis of the characteristics of the hypothesis ($|V_h|$ and $|E_h|$). If one reverses the roles of T and H, the normalizing factor will change.

$$entscore(T,H) = (\alpha \times \frac{\sum_{V_h \in H_v} \max_{V_t \in T_v} match(V_h, V_t)}{|V_h|} + \beta \times \frac{\sum_{E_h \in H_e} \max_{E_t \in T_e} match(E_h, E_t)}{|E_h|} + \gamma) \times (\frac{1+(-1)^{\#neg\_rel}}{2})$$

Equation 1. Scoring formula for graph subsumption.

4. CASE STUDIES

4.1. Two Examples

We present NLP challenges in two state-of-the-art ITSs that have been developed at The University of Memphis over the last decade: AutoTutor and iSTART. The two systems differ in their general mode of operation: AutoTutor is dialogue-based and iSTART is a response-based system.

4.2. AutoTutor

AutoTutor (autotutor.org) is an intelligent tutoring system with animated pedagogical agents (Graesser *et al*., 2005b). AutoTutor helps students learn about science and technology topics by holding a dialogue in natural language with the student. AutoTutor presents deep-reasoning questions to the student that call for about a paragraph of information in an ideal answer. AutoTutor helps students construct essay-like answers through multiple conversational turns with prompts, hints, feedback, and answers to student-questions via mixed-initiative dialogue.

AutoTutor integrates advances in discourse processes, computational linguistics, artificial intelligence, education, and multimedia (Graesser *et al*., 2004; Graesser, *et al*., 2005b; Graesser *et al*., 2001). AutoTutor has been developed for Newtonian physics, computer literacy, and critical thinking, but there are authoring tools for developing materials on new subject matters in a short amount of time (*i.e*., in a few weeks rather than a few years, as is the case for typical intelligent tutoring systems). AutoTutor's dialogues are organized around difficult questions that require reasoning and explanations. An ideal answer is about 3-7 sentences in length. The primary method of scaffolding good answers to questions is through expectation and misconception tailored dialogue. Both AutoTutor and human tutors (Graesser & Person, 1994) typically have a list of anticipated good answers (called

expectations, *e.g.*, force equals mass times acceleration) and a list of misconceptions associated with each main question. AutoTutor guides the student in articulating the expectations through a number of dialogue moves: pumps (what else?), hints, and prompts for specific information. As the learner expresses information over many turns, the list of expectations is eventually covered and the main question is scored as answered. Another conversation goal is to correct the misconceptions that are manifested in the student's talk. AutoTutor adaptively responds to the student by giving short feedback on the quality of student contributions (positive, negative or neutral) and by answering the student's questions. AutoTutor has been enhanced in a number of ways since its inception. Current versions of AutoTutor have speech recognition, guide students in using interactive simulations with microworlds, and respond to learner emotions that are detected through facial expressions, dialogue history, speech intonation, and body posture.

AutoTutor improves learning at deep levels of comprehension with effect sizes that vary between .20 and 2.30 (M = .80), depending on the subject matter, the test, and the comparison condition (*e.g.*, pretests and reading a textbook for an equivalent amount of time, Graesser *et al*., 2004; VanLehn *et al*., in press). AutoTutor is most effective when there is a large gap between the learner's prior knowledge and the ideal answers of AutoTutor.

It is beyond the scope of this chapter to address, at a deep level, all discourse issues involved in the dialogue between AutoTutor and a student. Instead, we focus on the specific issue of student answer evaluation at sentence level. During the student-AutoTutor interaction, the system gives hints for students to articulate the set of expectations, *i.e*. the ideal answer, to the problem. AutoTutor requires that the learner articulates each of the expectations before it considers the question answered. The system periodically identifies a missing expectation during the course of the dialogue and posts the goal of covering the expectation. For each student response to a hint, the system must evaluate whether the student answer matches the corresponding expectation. This is the student answer evaluation problem in AutoTutor. As an example, we consider the following expectation and student answer, reproduced from a previous experiment from AutoTutor (Graesser *et al*., 2005c).

E: *The person and the object cover the same horizontal distance.*
A: *The two objects will cover the same distance.*

AutoTutor uses mainly LSA or LSA combined with word overlap and Kendall's tau to evaluate sentential level student contributions. Our non-symmetric approach can be used in different ways to model the student answer evaluation problem in AutoTutor. Section 5 presents the details of the different models.

4.3. iSTART (Interactive Strategy Trainer for Active Reading and Thinking)

The primary goal of iSTART (iSTARTreading.com) is to help high school and college students learn to use reading comprehension strategies that support deeper understanding. Its design is based on a successful classroom intervention called Self-Explanation Reading Training (SERT; McNamara, 2004). SERT combines the power of self-explanation in facilitating deep learning (Chi *et al*., 1994) with content-sensitive, interactive strategy training (Bereiter & Bird; 1985). Specifically, students learn to self-explain using five reading strategies: *monitoring comprehension* (*i.e*., recognizing comprehension failures and the need for remedial strategies), *paraphrasing* the text, making *bridging inferences* between the current sentence and prior text, making *predictions* about the subsequent text, and *elaborating* the text with links to what the reader already knows. It uses animated conversational agents to scaffold the learning of these comprehension strategies (McNamara *et al*., 2007; McNamara *et al*., 2004; Millis *et al*., 2004).

Training with iSTART occurs in three phases and takes approximately 2½ hours. The first phase of training is strategy *Introduction*. This phase includes definitions and examples of self-explanation and the reading strategies. In the second phase of training, *Demonstration*, pedagogical agents model the use of the reading strategies while self-explaining a science text. Students identify the strategies used by the pedagogical agent. In the third phase, *Practice*, students read two short science passages and are asked to apply the newly learned strategies while typing self-explanations of the sentences in the texts. Scaffolded feedback is provided to the students based on computational algorithms that evaluate the quality of the explanations (McNamara *et al*., 2007; McNamara *et al*., 2004; Millis *et al*., 2004).

Studies have evaluated the impact of iSTART on both reading strategies and comprehension for thousands of high school and college students (McNamara *et al*., 2004). The results have revealed that reading strategies are facilitated by iSTART, with effect sizes that have vary between 0.35 and 1.00 (M = 0.68). Similarly, it has been shown that iSTART improves reading comprehension. The lower knowledge, less skilled students show the greatest benefit on text-based questions, with effect sizes that vary between 0.35 and 1.00 (M = 0.68). In contrast, higher knowledge, more skilled students showed the largest gains on the more complex bridging inferences questions, with effect sizes that vary between 1.04 and 1.20 (M = 1.12). Thus, students benefit from iSTART at their zone of proximal development, that is, the level of processing where they are ready to be scaffolded toward improvement.

In iSTART, we face similar, but not identical tasks to AutoTutor's sentence-level assessment tasks. During the final stage of the iSTART process, the student is given a sentence from a paragraph the student was exposed to earlier and asked to self-explain the sentence using one of several reading strategies. The reading

strategies include topic identification, paraphrasing, elaboration, logic or common sense, predictions, bridging. Details about the reading strategies can be found in (McNamara *et al.*, 2004). As an example of sentence level assessment in iSTART, we consider the following sentence, called Textbase (T), from a science textbook and the student input, called self-explanation (SE), reproduced from a recent iSTART experiment. The SE is reproduced as typed by the student. The SE below is a paraphrase of the text T.

T: *Plants and algae have an additional kind of energy-converting organelle, called a chloroplast*.
SE: *chloroplast is an kind of energy-converting organelle.*

Similar to the student answer evaluation problem in AutoTutor, we focus here on assessing the SE to the original text T and do not address discourse or domain-specific understanding.

Currently, iSTART relies on a hybrid approach that combines LSA and content word overlap adjusted for sentence length to interpret sentential self-explanations of learners. Several ways to model each reading strategy in iSTART using our LSE approach are possible. Section 5 gives details about these models.

5. MODELLING SENTENCE-LEVEL ASSESSMENT TASKS IN ITSS

We presented earlier two fundamental problems in AutoTutor and iSTART: the student answer evaluation problem in AutoTutor and assessing self-explanations in iSTART. There are many similarities among these problems. The two problems are a critical processing step in the overall processing scheme of the corresponding ITSs. They all deal with an ideal text (Expectation and Text, respectively) and a student articulated text (Student Answer and Self-Explanation, respectively). The two texts are typically the size of a sentence. The main task is to assess what is the relationship between the two texts. Depending on the goal of the ITS, the interpretation of the assessment outcome could be a judgment as to whether the two texts are a paraphrase of each other or whether one text is an elaboration of the other. Depending on the type of relationship between the texts, we model the assessment problem differently using our proposed LSE approach for entailment (Rus & Graesser, 2006; Rus *et al*., 2008, 2009).

The student answer evaluation problem in AutoTutor can be modeled as an entailment, reverse entailment, or paraphrase problem. Reverse entailment is the problem in which we check whether the Hypothesis entails the Text. This situation could occur when the Hypothesis is longer, in number of words, than the Text, (*e.g*. elaborations in iSTART).

Evaluating student answers in AutoTutor is analogous to an entailment problem. To illustrate, consider the *Pumpkin* problem from the AutoTutor library:

Suppose a runner is running in a straight line at constant speed, and the runner throws a pumpkin straight up. Where will the pumpkin land? Explain why. An expectation and real student answer typed by a participant in a prior experiment are give below.

Expectation: *The object will continue to move at the same horizontal velocity as the person when it is thrown.*
Student Answer: *The pumpkin and the runner have the same horizontal velocity before and after release.*

Such expectation-student answer pairs can be viewed as an entailment pair of Text-Hypothesis, respectively, and can be evaluated as T or F. The task is to find the truth value of the student answer based on the true fact encoded in the expectation.

The student answer evaluation task can also be modeled as reverse entailment. That is, we assume that the student answer is true and try to see if it entails the expectation. In this case, we can compute an entailment score between the student answer and expectation, *i.e*. entscore (Student Answer, Expectation). Lastly, the problem can be modeled as a paraphrase. We want to check if the student answer is just a paraphrase of the ideal answer. In this case we can use the average score of standard and reverse entailment.

Self-Explanations in iSTART must be assessed with respect to several sentence-to-sentence relations such as elaboration, paraphrase, topic identification, or bridging inference. Each such relation needs individual modeling.

Paraphrase responses are restatements of the Text, incorporating different words and syntax while lacking any kind of frozen expressions. The paraphrase task will be modeled as two entailment tasks (1) between the textbook sentence and self-explanation and (2) vice versa. iSTART distinguishes among several types of paraphrases. When the student self-explanation is very close in terms of words and structure to the original textbook sentence, we have a *paraphrase-close*. iSTART does not values such paraphrases as good reading strategies. A *paraphrase-distant* self-explanation is highly similar to the original sentence in terms of semantics but different in terms of structure and/or content words. *Paraphrase-Inaccurate* sentences were defined as a failed paraphrase. For example, a participant may have used similar words to the target sentence but created a sentence with a different meaning.

Text: *Ribosomes are the smallest but most abundant organelles.*
Paraphrase-distant: *one of the most abundant organelles is called the ribosomes and it is the known smallest organelles.*

Text: *The nucleus has often been called the control center of the cell.*
Paraphrase-close: *the nucleus can also be called the control center of the cell.*

Text: *Almost every chemical reaction that is important to the cell's life involves some kind of protein.*

Paraphrase-inaccurate: *That means proteins got something to do with chemical reactions.*

Elaboration should be modeled as a reverse-entailment problem because usually an elaborated self-explanation is longer than the original text sentence and thus it is the case that the self-explanation subsumes the original text sentence.

Text: *These amino acids are hooked together to make proteins at very small organelles called ribosomes.*
Elaboration: *the amino acids which is a chemical are hooked to nucleolus to make a protein called ribosomes thats very small.*

Topic identification is modeled using reverse entailment score because such responses tend to include what the sentence was about. Thus, sentences often began with frozen expressions such as "The sentence talks about …". The assumption is that the self-explanation will subsume the textbook sentence in which case reverse entailment is the best model.

Text: *The largest and most visible organelle in a eukaryotic cell is the nucleus.*
Topic ID: This is about large and most visible organelle in eukaryotic cell is the nucleus.

Bridging inference is a complex reading strategy in iSTART in which the student self-explains a target sentence using additional information from a previous sentence. One example of a bridging inference is given below. The student's self-explanation contains a bridging inference between the second sentence below, *i.e.* the target sentence the student was asked to self-explain, and a previous sentence in the textbook paragraph, the first sentence below.

1. Though the queen is the most important single individual in honeybee society, she in no way rules the hive.
2. She does produce hormones that control various aspects of bee behavior.
Bridging inference: although the queen doesn't rule, she does produce hormones that effect bee behavior.

Bridging inferences can be modeled as a standard entailment problem. There is the extra computational challenge to generate a graph for the several sentences that form the entailing text. Resolving anaphors/coreferences between these sentences could lead to a single entailing text graph that can be used to check whether it subsumes the self-explanation graph.

6. RESULTS

A prototype of the Lexico-Syntactic Entailment (LSE) approach has been evaluated on data from the *recognizing textual entailment* task (RTE; Dagan *et al*., 2005) and data from two ITSs, AutoTutor and iSTART. Our studies have shown that LSE delivers high performance analyses when compared to similar systems in the industry approved testing ground of the RTE tasks (Rus *et al*., 2006; Rus *et al*., 2005a). However, the natural language input from ITSs (with its spelling, grammar, asymmetrical, and syntax issues) provides a far sterner testing ground. Results from these studies suggest that in this environment also, the performance of LSE has been significantly better than comparable approaches and often at least equal to that of human raters (see McCarthy *et al*., 2007b).

Results on RTE Data. The RTE challenge offers development and test data for entailment evaluations. The Text-Hypothesis pairs in RTE data sets are collected from several natural language processing applications including Information Retrieval, Question Answering, and Summarization. Both the Text and the Hypothesis are sentential text fragments. Each pair is labeled with True or False, depending on whether or not the Text entails the Hypothesis. RTE data sets are balanced (50–50 split) with respect to the number of True and False cases. In several studies (Rus *et al.* 2005a; Rus *et al*., 2005b), the LSE approach was significantly more accurate than the baseline of randomly guessing (random baseline) True or False ($p < .01$): accuracy = 0.553 and CWS = 0.604 (Confidence Weighted Score ranges from 0 [= no correct judgments] to 1 [= perfect score], with higher scores indicating greater confidence in the correctness of the judgments, Dagan *et al*., 2005). The Baselines row in Table 2 shows values for the random baseline as well as a uniform baseline that consistently predicts True or False (Dagan *et al*., 2005). For comparison purposes, Table 2 replicates the results of similar systems that participated in RTE-1 (see Table 2 in Dagan *et al*., 2005). An LSA-based approach did not yield significant results on RTE test data (CWS = 0.5122; accuracy = 0.5050). Among the systems in Table 2, only LSE provided significant results above the random baseline at 0.01 level.

Table 2. Results on RTE-1 test data set for systems using similar resources to LSE

System	CWS	Accuracy
LSE	0.604	0.553
Zanzotto (Rome-Milan)	0.557	0.524
Punyakanok	0.519	0.515
Andreevskaia	0.519	0.515
Jijkoun	0.553	0.536
Baselines	0.540*/0.558**/0.500***	0.535*/0.546**/0.500***

Note: * p < .05 and ** p < .01 random baseline; *** uniform baseline

When compared to similar systems in RTE-2 (Bar-Haim *et al*., 2006), our system yielded better results (accuracy = 0.590 and CWS = 0.604) than similar systems and even better results over systems (see the systems Delmonte [accuracy = 0.547 and CWS=0.549] and Kozareva [accuracy = 0.550 and CWS = 0.548] in Bar-Haim *et al*., 2006) that use more resources, *e.g*., paraphrase templates, than our LSE approach.

Results on ITS Data. Rus and Graesser (2006) used expert evaluations from physicists on a test set of 125 expectation-student answers pairs collected from a sample of AutoTutor tutorial dialogues on Newtonian physics. The object of the experiment was to compare LSE to LSA using the expert-evaluated expectation-student answer pairs. The LSE approach provided an accuracy of 0.690, whereas LSA yielded 0.600. Such a result illustrates the value of augmenting AutoTutor with lexico-syntactic natural language understanding. When evaluation decisions for CWS were taken into consideration, a word overlap approach enhanced with synonymy provided best results (0.800 as compared to 0.766 for the LSE approach and 0.597 for LSA), which shows the need for customization. LSE without synonymy provided the best precision results (0.866) compared with word overlap and synonymy (0.822) and LSA (0.673) approaches, indicating that syntax combined with simple word overlap but no synonymy leads to high-precision solutions.

In Rus *et al*. (2007), a corpus of iSTART self-explanation responses (357) was evaluated by an array of textual evaluation measures. The study compared the proposed LSE approach with several lexical-based approaches, including LSA and word overlap, on the task of distinguishing two categorizations: *good paraphrases* and *bad paraphrases* (labeled as *TopicID*). The results demonstrated that the LSE approach was the most powerful distinguishing index of the self-explanation categories (LSE: $F(1,1228) = 25.05$, $p < .001$; LSA: $F(1,1228) = 2.98$, $p > .01$). The final model (generated from a discriminant analysis) resulted in a significant classification of the two groups (TopicID category: recall = .692; precision = .409; Paraphrase category: recall = .743; precision = .904). Although the accuracy of the model was significant ($X^2 = 17.27$, $df = 1$, $p < .001$) and encouraging, Rus and colleagues reported in *post hoc* analysis that the lower accuracy of TopicID category was probably caused by the contribution to the model of LSA: specifically, the LSA values were more likely to misclassify results because longer sentences were evaluated too highly.

In McCarthy *et al*. (2007a), iSTART self explanations were hand-coded for degree of entailment, paraphrase, and elaboration. In a series of multiple regression tests, the entailment evaluation once again proved to be a more powerful predictor of these categories than traditional measures: for entailment, the LSE approach was a significant predictor ($t = 9.61$, $p<.001$) and LSA was a marginal predictor ($t = -1.90$, $p = .061$); for elaboration and for paraphrase the LSE approach was again a significant predictor ($t = -7.98$, $p < .001$; $t = 5.62$, $p < .001$, respectively), whereas LSA results were not significant.

McCarthy *et al*. (2007b) compared the LSE approach, to a variety of other text relatedness metrics (LSA, content-overlap, and edit distances). Our corpus was formed from 631-sentence self-explanations from a recent iSTART experiment. The study used the values of student response inputs that had been hand coded by experts across three categories of text relatedness: paraphrase, entailment, and elaboration. A series of regression analyses suggested that the LSE approach was the best measure for approximating these hand coded values. A reverse entailment score explained approximately 0.670 of the variance for paraphrase; the standard LSE index explained approximately 0.550 of the variance for entailment, and 0.450 of the variance for elaboration. The derived evaluations either met or surpassed human inter-rater correlations, suggesting that the LSE approach can produce assessments of text at least equal to that of expert raters.

7. CONCLUSIONS

We presented an overview of computational challenges regarding natural language input in Intelligent Tutoring Systems. In particular, we discussed the requirements of approaches to the task of assessing natural language student input in tutoring systems. Approaches need to be accurate enough to provide appropriate evaluation, fast enough to operate in real time, and flexible enough to be easily ported. Such an approach, lexico-syntactic entailment (LSE), was presented in detail.

Our studies demonstrated that LSE has a number of advantages over currently available approaches. First, LSE results are a significant improvement over existing lexical-based approaches used in ITSs. Further, LSE results are comparable to the gold standard of human raters, in some cases significantly surpassing inter-rater agreement. Second, LSE evaluation is non-symmetrical, entscore(H, T) ≠ entscore(T,H). As such, the approach can be easily extended to capture a wider variety of relations among text fragments. For instance, LSE can be adapted for handling paraphrases: If text T1 is a paraphrase of text T2 then T1 should entail T2 and T2 should entail T1. Based on this observation, a paraphrase score can be defined as the average of entscore(T1, T2) and entscore(T2, T1). This type of model is called *symmetrical*. Reverse entailment (where we assess the degree to which the Hypothesis subsumes the Text) is also possible. This situation could occur when the Hypothesis contains more words than the Text, as is common for elaborative responses in iSTART. Third, LSE includes a confidence evaluation. The confidence evaluation is beneficial because it assists in selecting appropriate feedback to students. For instance, two student inputs that are assessed as close to the stored case/ideal answer may lead to different feedback depending on the confidence on that assessment. If there is high confidence that one input is close to the ideal answer then positive feedback should be offered. If the confidence in the closeness decision is low then neutral feedback could be provided. Fourth, LSE is *customizable*. That is, some components of the LSE approach can be turned on and

off or given a small weight. For instance, the syntactic component can be turned off for tasks that are known to be handled better with just lexical information. Fifth, LSE offers more interactivity and scalability potential than logic-based approaches. The greater interactivity stems from its reliance on fewer resources to interpret user input. The LSE approach only uses a syntactic parser and WordNet based synonyms and antonyms. It does not use *logic provers* or heavier components that require greater computational resources. Furthermore, the approach is more scalable than logic-based approaches because porting it from one subject area to another does not require redesign of any of the components. Sixth, the LSE approach addresses the common problems associated with textual assessment and understanding outlined in Section 3. In particular, the text length confound did not occur in our experiments. Finally, it is important to note that the iSTART and AutoTutor studies are highly encouraging because they use uncorrected, unedited, student responses as their source of input. We expect these results to improve further.

REFERENCES

1. ANDERSON J. R., CORBETT A. T., KOEDINGER K., and PELLETIER R., Cognitive tutors: Lessons learned. *The Journal of Learning Sciences, 4*, 167-207, 1995.
2. BAR-HAIM R., DAGAN I., DOLAN B., FERRO L., GIAMPICCOLO D., MAGNINI B., and SZPEKTOR I., The second PASCAL recognizing textual entailment challenge, in *Proceedings of the Second PASCAL Challenges Workshop on Recognizing Textual Entailment.* Venice, Italy 2006.
3. BEREITER C. and BIRD M., Use of thinking aloud in identification and teaching of reading comprehension strategies. *Cognition and Instruction*, 2, 131-156, 1985.
4. BURSTEIN J., The e-rater scoring engine: Automated essay scoring with natural language processing, in M.D. Shermis & J. Burstein (eds.): *Automated essay scoring: A cross-disciplinary perspective*. Hillsdale, NJ: Erlbaum, 2003.
5. CHI M. T. H., DE LEEUW N., CHIU M. H., and LAVANCHER C., Eliciting self-explanations improves understanding, *Cognitive Science, 18*, 439-477, 1994.
6. DAGAN I., GLICKMAN O., and MAGNINI B., The PASCAL recognizing textual entailment challenge, in *Proceeding of the First PASCAL Challenges Workshop on Recognizing Textual Entailment*. Southampton, UK: Pattern Analysis, Statistical Modeling and Computational Learning, Inc., 2005.
7. DENNIS S., Introducing word order in a LSA framework, in T. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (eds.), *Handbook on Latent Semantic Analysis.* Erlbaum, 2007.
8. DODDS P. and FLETCHER J. D., Operations for new "smart" learning environments enabled by next-generation web capabilities, *Journal of Education Multimedia and Hypermedia, 13*, 391-404, 2004.
9. FLETCHER J. D., Evidence for learning from technology-assisted instruction, in H. F. O'Neil, Jr., R. Perez (eds.), *Technology applications in education: A learning view* (pp.79-99). Hillsdale, NJ: Lawrence Erlbaum, 2003.
10. FOLTZ W., GILLIAM S., and KENDALL S., Supporting content-based feedback in on-line writing evaluation with LSA. *Interactive Learning Environments*, 8, 111-128, 2000.
11. FOLTZ P. W., Latent Semantic Analysis for text-based research, *Behavior Research Methods, Instruments and Computers*. 28(2), 197-202, 1996.
12. FREEDMAN R. K., ROSE C. P., RINGENBERG M. A., and VANLEHN K., ITS Tools for Natural Language Dialogue: A Domain Independent Parser and Planner, in *Proceedings of the Intelligent Tutoring Systems Conference*, 2000.

13. GRAESSER A. C., PENUMATSA P., VENTURA M., CAI Z., and HU X., Using LSA in AutoTutor: Learning through mixed initiative dialogue in natural language, in T. Landauer, D. McNamara, S. Dennis, and W. Kintsch (eds.): *LSA: A Road to meaning*. Mahwah, NJ: Erlbaum, 2007.
14. GRAESSER A. C., CHIPMAN P., HAYNES B. C., and OLNEY A., AutoTutor: An intelligent tutoring system with mixed-initiative dialogue. *IEEE Transactions in Education*, 48, 612-618, 2005.
15. GRAESSER A.C., MCNAMARA D.S., and VANLEHN K., Scaffolding deep comprehension strategies through Point&Query, AutoTutor, and iSTART. *Educational Psychologist*, 40, 225-234, 2005b.
16. GRAESSER A.C., HU X., and MCNAMARA D.S., Computerized learning environments that incoporate research in discourse psychology, cognitive science, and computational linguistics, in A.F. Healy (ed.): *Experimental Cognitive Psychology and its Applications: Festschrift in Honor of Lyle Bourne, Walter Kintsch, and Thomas Landauer*. Washington, D.C., American Psychological Association, 2005c.
17. GRAESSER A. C., LU S., JACKSON G. T., MITCHELL H., VENTURA M., OLNEY A., and LOUWERSE M. M., AutoTutor: A tutor with dialogue in natural language. *Behavioral Research Methods, Instruments, and Computers, 36*, 180-192, 2004.
18. GRAESSER A.C., VANLEHN K., ROSE C., JORDAN P., and HARTER D., Intelligent tutoring systems with conversational dialogue. *AI Magazine, 22*, 39-51, 2001.
19. GRAESSER A. C. and PERSON N. K., Question asking during tutoring. *American Educational Research Journal, 31*, 104-137, 1994.
20. GRAESSER A.C., SINGER M., and TRABASSO T., Constructing inferences during narrative text comprehension. *Psychological Review*, 101, 371-95, 1994
21. HOBBS J. R., STICKEL M., MARTIN P., and EDWARDS D., Interpretation as abduction, in *Proceedings of the 26th Meeting of the ACL*, Association of Computational Linguistics, pp.95-103, Buffalo, New York, 1988.
22. JORDAN P. W. and VANLEHN K., Discourse Processing for Explanatory Essays in Tutorial Applications, *Proceedings of the 3rd SIGdial Workshop on Discourse and Dialogue*, pp. 74-83, 2002.
23. JURAFSKY D and. MARTIN J.H., *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice-Hall, 2000.
24. KINTSCH E., CACCAMISE D., FRANZKE M., JOHNSON N., and DOOLEY S., Summary street: LSA-Based software for comprehension and writing. In D. S. McNamara, T. Landauer, W. Kintsch, and S. Dennis (eds.): *Handbook of Latent Semantic Analysis*, Mahwah, NJ: Erlbaum, 2007.
25. KINTSCH W., *Comprehension: A paradigm for cognition*. New York: Cambridge University Press, 1998.
26. KINTSCH W., Information accretion and reduction in text processing: Inferences. *Discourse Processes*, 16, 193-202, 1993.
27. KOEDINGER K. R., ANDERSON J. R., HADLEY W. H., and MARK M. A., Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, 8, 30-43, 1997.
28. LANDAUER T. K., MCNAMARA D. M., DENNIS S., and KINTSCH W., *Handbook of latent semantic analysis.* Mahwah, NJ: Erlbaum, 2007.
29. LESGOLD A., LAJOIE S. P., BUNZO M., and EGGAN G. SHERLOCK, A coached practice environment for an electronic trouble-shooting job, in J. H. Larkin & R. W. Chabay (eds.): *Computer assisted instruction and intelligent tutoring systems: Shared goals and complementary approaches* (pp.201-238). Hillsdale, NJ: Erlbaum, 1992.
30. MARCUS M., SANTORINI B., and MARCINKIEWICZ M. A., Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics, 19*, 313-330, 1993.
31. MATAKCHEV M., JORDAN P. W., and VANLEHN K., Abductive theorem proving for analyzing student explanations to guide feedback in intelligent tutoring systems, *Journal of Automated Reasoning, Special Issue on Automated Reasoning and Theorem Proving in Education*, 32, 187-226, 2004.
32. MCCARTHY P. and JARVIS S., *vocd*: A theoretical and empirical evaluation. *Language Testing*, 24, 459-488, 2007.

33. MCCARTHY P. M., RUS V., CROSSLEY S. A., BIGHAM S. C., GRAESSER A. C., and MCNAMARA D. S., Assessing entailer with a corpus of natural language, in D. Wilson and G. Sutcliffe (eds.): *Proceedings of the twentieth International Florida Artificial Intelligence Research Society Conference* (pp. 247-252). Menlo Park, California: The AAAI Press, 2007a.
34. MCCARTHY P. M., RUS V., CROSSLEY S. A., GRAESSER A. C., and MCNAMARA D. S., Providing a feedback facility to an Intelligent Tutoring System using multiple text analysis measures, in *Submission to the Florida Artificial Intelligence Research Society, 2008*, 2007b.
35. MCNAMARA D. S., BOONTHUM C., LEVINSTEIN I. B., and MILLIS K., Evaluating self-explanations in iSTART: Comparing word-based and LSA analysis, in T. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (Eds.), *Handbook of latent semantic analysis* (pp. 227-242). Mahwah, NJ: Erlbaum, 2007.
36. MCNAMARA D.S., OZURU Y., GRAESSER A.C., and LOUWERSE M., Validating Coh-Metrix, in R. Son (ed.): *Proceedings of the 28th Annual Meetings of the Cognitive Science Society*. (pp. 573-578). Mahwah, NJ: Erlbaum, 2006.
37. MCNAMARA D. S., SERT: Self-explanation reading training. *Discourse Processes*, 38, 1-30, 2004.
38. MCNAMARA D. S., LEVINSTEIN I. B., and BOONTHUM C., iSTART: Interactive strategy trainer for active reading and thinking, *Behavior Research Methods, Instruments, and Computers*, 3b, 222-233, 2004.
39. MEL'CUK I., *Dependency syntax: Theory and practice*, Albany, US: State University of New York Press, 1998.
40. MILLER G.A., WordNet: a lexical database for English, *Communications of the ACM*, v.38 n.11, p.39-41, Nov. 1995.
41. MILLIS K., MAGLIANO J., WIEMER-HASTINGS K., TODARO S., and MCNAMARA D. S., Assessing and improving comprehension with Latent Semantic Analysis, in T. Landauer, D. S. McNamara, S. Dennis, & W. Kintsch (eds.): *Handbook of Latent Semantic Analysis*, pp. 207-225, Mahwah, NJ: Erlbaum, 2007
42. MILLIS K., KIM H. J., TODARO S., MAGLIANO J. P., WIEMER-HASTINGS K., and MCNAMARA D. S., Identifying reading strategies using latent semantic analysis: Comparing semantic benchmarks. *Behavior Research Methods, Instruments, & Computers*, 36, 213-221, 2004.
43. MITROVIC A., SURAWEERA P., MARTIN B., and WEERASINGHE A., DB-suite: experiences with three intelligent, web-based database tutors. *Journal of Interactive Learning Research*, 15, 409-432, 2004.
44. MOLDOVAN D. and RUS V., Logic form transformation of wordnet and its applicability to question answering, in *Proceedings of the ACL Conference (ACL-2001)*, 2001.
45. MONZ C. and DE RIJKE M., Light-Weight Entailment Checking for Computational Semantics, in P. Blackburn and M. Kohlhase (eds.): *Proceedings of the 3rd Workshop on Inference in Computational Semantics (ICoS-3)*, pp. 59–72, 2001.
46. PAZIENZA M., PENNACCHIOTTI M., and ZANZOTTO F., Textual entailment as syntactic graph distance: A rule based and SVM based approach, in *Proceedings of the RTE Challenge Workshop*, pp. 25–28. Southampton, UK, 2005.
47. PENUMATSA P., VENTURA M., GRAESSER A.C., FRANCESCHETTI D.R., LOUWERSE M., HU X., CAI Z., & the Tutoring Research Group, The right threshold value: What is the right threshold of cosine measure when using latent semantic analysis for evaluating student answers? *International Journal of Artificial Intelligence Tools*, 12, 257-279, 2004.
48. REHDER B. and HASTIE R., The differential effects of causes on categorization and similarity. In *The Proceedings of the Twentieth Annual Conference of the Cognitive Science Society* (pp. 893-898). Madison, WI, 1998.
49. ROSE C. P., A Framework for Robust Semantic Interpretation, in *Proceedings of 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pp. 311-318, Seattle, Washington, USA, 2000.
50. RUS V., MCCARTHY P. M., GRAESSER A. C., and MCNAMARA D. S., Identification of Sentence-to-Sentence Relations Using a Textual Entailer. *The International Journal on Research in Language and Computation*, 7, pp. 1-21, November 2009.

51. RUS V., MCCARTHY P.M., MCNAMARA D.S., and GRAESSER A.C., Natural Language Understanding and Assessment, in J.R. Rabunal, J. Dorado, A. Pazos (eds.): *Encyclopedia Artificial Intelligence. Information Science Reference*, Hershey, PA, 2008.
52. RUS V., MCCARTHY P. M., LINTEAN M. C., GRAESSER A. C., and MCNAMARA D. S., Assessing student self-explanations in an intelligent tutoring system, in D. S. McNamara and G. Trafton (eds.): *Proceedings of the 29th annual conference of the Cognitive Science Society*, pp 623-628, Cognitive Science Society, 2007.
53. RUS V. and GRAESSER A.C., Deeper natural language processing for evaluating student answers in intelligent tutoring systems, in the *Proceedings of the American Association of Artificial Intelligence*, pp. 1495-1500, Menlo Park, CA: AAAI, 2006.
54. RUS V., MCCARTHY P.M., and GRAESSER A.C., Analysis of a Textual Entailer, in Alexander Gelbukh (ed.): *Computational Linguistics and Intelligent Text Processing, 7th International Conference, CICLing 2006*, pp. 287-298, Mexico City, Mexico, Lecture Notes in Computer Science 3878 Springer Verlag, 2006.
55. RUS V. and DESAI K., Assigning functional tags with a simple model, in Alexander Gelbukh (ed.), *Computational Linguistics and Intelligent Text Processing, 6th International Conference, CICLing 2005*, pp. 112-115, Mexico City, Mexico, Lecture Notes in Computer Science 3406, Springer Verlag, 2005
56. RUS V., GRAESSER A., and DESAI K., Lexico-Syntactic Subsumption for Textual Entailment. In Nicolas Nicolov, Kalina Bontcheva, Galia Angelova and Ruslan Mitkov (eds.): *Recent Advances in Natural Language Processing IV. Selected papers from RANLP 2005*, Current Issues in Linguistic Theory 292, John Benjamins Publishing Company (2007), pp. 187-196, 2005a.
57. RUS V., GRAESSER A.C., MCCARTHY P. M., and LIN K., A Study on Textual Entailment. In *Proceedings of the IEEE's International Conference on Tools with Artificial Intelligence (ICTAI 2005)*, pp.326-333, November 14-16, 2005, Hong Kong, 2005b.
58. RUS V., *Logic Form For WordNet Glosses and Application to Question Answering*, Computer Science Department, School of Engineering, Southern Methodist University, PhD Thesis, May 2002, Dallas, Texas, 2002.
59. RUS V., High Precision Logic Form Transformation, in *Proceedings of the 13th IEEE International Conference on Tools with Artificial Intelligence (ICTAI)* 2001, pp.288, Dallas, TX, 2001.
60. TVERSKY A., Features of similarity, *Psychological Review*, 84, 327-352, 1977.
61. TWEEDIE F. J. and BAAYEN R. H., How variable may a constant be? Measures of lexical richness in perspective, *Computers and the Humanities* 32, 323–352, 1998.
62. VANLEHN K., GRAESSER A. C., JACKSON G. T., JORDAN P., OLNEY A., and ROSE C. P., When are tutorial dialogues more effective than reading? *Cognitive Science*, 31, 3-62, 2007.
63. VANLEHN K., LYNCH C., SCHULTZ K., SHAPIRO J. A., SHELBY R. H., TAYLOR L., TREACY D. J., WEINSTEIN A., and WINTERSGILL M. C., The Andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence and Education*, 15, 147-204, 2005.
64. VANLEHN K., JORDAN P., ROSÉ C. P., BHEMBE D., BOTTNER M., GAYDOS A., MAKATCHEV M., PAPPUSWAMY U., RINGENBERG M., ROQUE A., SILER S., and SRIVASARTE R., The architecture of why2-atlas: A coach for qualitative physics essay writing. S. A. Cerri, G. Gouarderes, and F. Paraguacu (eds.): *Proceedings of Intelligent Tutoring Systems Conference*, pp.158-167, Springer, 2002.
65. WIEMER-HASTINGS P., WIEMER-HASTINGS K., and GRAESSER A.C., Improving an intelligent tutor's comprehension of students with Latent Semantic Analysis, *Artificial Intelligence in Education*, pp. 535—542, IOS Press, 1999.
66. WISHER R. A. and FLETCHER J. D., The case of advanced disturbed learning, *Information and Security: An International Journal*, 14, 17-25, 2004.
67. WOLFE M., B. SCHREINER M. E., REHDER B., LAHAM D., FOLTZ P. W., KINTSCH W., and LANDAUER T. K., Learning from text: Matching readers and texts by latent semantic analysis. *Discourse Processes, 25*, 309-336, 1998.