Automatic Evaluation of Students' Answers using Syntactically Enhanced LSA

Dharmendra Kanejiya[*] and **Arun Kumar**[†] and **Surendra Prasad**[*]
[*]Department of Electrical Engineering
[†]Centre for Applied Research in Electronics
Indian Institute of Technology
New Delhi 110016 INDIA
kanejiya@hotmail.com arunkm@care.iitd.ernet.in sprasad@ee.iitd.ernet.in

Abstract

Latent semantic analysis (LSA) has been used in several intelligent tutoring systems(ITS's) for assessing students' learning by evaluating their answers to questions in the tutoring domain. It is based on word-document co-occurrence statistics in the training corpus and a dimensionality reduction technique. However, it doesn't consider the word-order or syntactic information, which can improve the knowledge representation and therefore lead to better performance of an ITS. We present here an approach called Syntactically Enhanced LSA (SELSA) which generalizes LSA by considering a word along with its syntactic neighborhood given by the part-of-speech tag of its preceding word, as a unit of knowledge representation. The experimental results on AutoTutor task to evaluate students' answers to basic computer science questions by SELSA and its comparison with LSA are presented in terms of several cognitive measures. SELSA is able to correctly evaluate a few more answers than LSA but is having less correlation with human evaluators than LSA has. It also provides better discrimination of syntactic-semantic knowledge representation than LSA.

1 Introduction

Computer based education systems are useful in distance learning as well as for class-room learning environment. These systems are based on intelligent tutoring systems(ITS's) which provide an interactive learning environment to students. These systems first familiarize a student with a topic and then ask questions to assess her knowledge. Automatic evaluation of students' answers is thus central to design of an ITS that can function without the need of continuous monitoring by a human. Examples of ITS's that use natural language processing to understand students' contribution are CIRCSIM (Glass, 2001), Atlas (Freedman et al., 2000), PACT (Aleven et al., 2001) etc. These systems use a parser to derive various levels of syntactic and semantic information and rules to determine the next dialog move. They perform quite well with short answers in a limited domain, but are limited to take arbitrarily long free-text input and are difficult to port across domains. These limitations can be alleviated by using latent semantic analysis(LSA), a recently developed technique for information retrieval (Deerwester et al., 1990), knowledge representation (Landauer et al., 1998), natural language understanding and cognitive modeling (Graesser et al., 1999; Graesser et al., 2000) etc. LSA has been used in various ITS's like AutoTutor (Wiemer-Hastings et al., 1998), Intelligent Essay Assessor (Foltz et al., 1999), Summary Street (Kintsch et al., 2000), Apex (Dessus et al., 2000) etc.

LSA is a statistical corpus-based natural language understanding technique that supports semantic similarity measurement between texts. Given a set of documents in the tutoring domain, LSA uses the frequency of occurrence of each word in each document to construct a word-document co-occurrence matrix. After preprocessing, singular value decomposition is performed to represent the domain knowledge into a 200 to 400 dimensional space. This space is then used for evaluating the semantic similarity between any two text units.

In an ITS, LSA is used to evaluate students' answers with respect to the ideal answers to questions in the domain (Graesser et al., 2000). This is done by finding the match between a student's answer and the ideal answer by calculating the cosine similarity measure between their projections in LSA space. This information is used to provide interactive response to the student in terms of hint, prompt,question etc.

It has been found that LSA performs as good as an intermediate expert human evaluator but not so well as an accomplished expert of the domain. This may be because LSA is a 'bag-of-words' approach and so lacks the word-order or syntactic information in a text document. But for correct automatic evaluation of students' answers, a model should consider both syntax and semantics in the answer. So, one obvious way to improve the performance of LSA is to incorporate some syntactic information in it.

In order to add syntactic information to LSA, recently there has been an effort in (Wiemer-Hastings and Zipitria, 2001), where a word along with its part-of-speech (POS) tag was used to construct the LSA matrix, thus capturing multiple syntactic senses of a word. But this approach, called *tagged LSA*, deteriorated the performance. In another attempt (Wiemer-Hastings and Zipitria, 2001), similarity between two sentences was calculated by averaging the LSA based similarity of sub-sentence structures like noun phrase, verb phrase, object phrase etc. This approach, called as *structured LSA* (SLSA), could improve the performance in terms of sentence-pair similarity judgment. But its performance in terms of evaluating students' answers was poorer than that of LSA(Wiemer-Hastings, 2000).

We propose here a model called *Syntactically Enhanced LSA* (SELSA), where we augment each word with the part-of-speech (POS) tag of the preceding word. Thus instead of word-document co-occurence matrix, we generate a matrix in which rows correspond to all possible word - POS tag combinations and columns correspond to documents. A preceding tag indicates some kind of syntactic neighbourhood around the focus word. Depending on the preceding tag, the syntactic-semantic sense of a word can vary. Thus SELSA captures finer resolution of syntactic-semantic information compared to mere semantics of LSA. This finer information can therefore be used to evaluate a student's answer more accurately than LSA.

We compare the performance of SELSA with LSA for the AutoTutor cognitive modeling task (Graesser et al., 1999). This involves evaluating students' answers to questions in three areas of computer science *viz.* hardware, operating system and networking. The performance is measured in terms of various criteria like correlation, mean absolute difference and number of correct /emphvs false evaluations by humans and by computer. SELSA is found better than LSA in terms of robustness across thresholds as well as in terms of evaluating more answers correctly, but it is having less correlation measure with human than LSA.

The organization of this paper is as follows. The next section describes LSA and its applications in ITS's. In section 3, we describe the proposed SELSA model. The experimental details are given in section 4 followed by discussion on results in section 5.

2 LSA in Intelligent Tutoring Systems

2.1 A Brief Introduction to LSA

LSA is a statistical-algebraic technique for extracting and inferring contextual usage of words in documents (Landauer et al., 1998). A document can be a sentence, a paragraph or even a larger unit of text. It consists of first constructing a word-document co-occurrence matrix, scaling and normalizing it with a view to discriminate the importance of words across documents and then approximating it using singular value decomposition(SVD) in R dimensions (Bellegarda, 2000). It is this dimensionality reduction step through SVD that captures mutual implications of words and documents and allows us to project any text unit whether a word, a sentence or a paragraph as a vector on the latent "semantic" space. Then any two documents can be compared by calculating the cosine measure between their projection vectors in this space.

LSA has been applied to model various ITS related phenomena in cognitive science e.g. judgment of essay quality scores (Landauer et al., 1998), assessing student knowledge by evaluating their answers to questions etc (Graesser et al., 2000), deciding tutoring strategy (Lemaire, 1999). It has been also used to derive a statistical language model for large vocabulary continuous speech recognition task (Bellegarda, 2000).

2.2 LSA based ITS's

Researchers have long been attempting to develop a computer tutor that can interact naturally with students to help them understand a particular subject. Unfortunately, however, language and discourse have constituted a serious barrier in these efforts. But recent technological advances in the areas of latent semantic processing of natural language, world knowledge representation, multimedia interfaces etc have made it possible for various teams of researchers to develop ITS's that approach human performance. Some of these are briefly reviewed below.

2.2.1 AutoTutor

AutoTutor task (Graesser et al., 1999) was developed at Tutoring Research Group of University of Memphis. AutoTutor is a fully automated computer tutor that assists students in learning about hardware, operating systems and the Internet in an introductory computer literacy course. AutoTutor presents questions and problems from a curriculum script, attempts to comprehend learner contributions that are entered by keyboard, formulates dialog moves that are sensitive to the learner's contributions (such as prompts, elaborations, corrections and hints), and delivers the dialog moves with a talking head. LSA is a major component of the mechanism that evaluates the quality of student contributions in the tutorial dialog. It was found that the performance of LSA in terms of evalu-

ating answers from college students was equivalent to an intermediate expert human evaluator.

2.2.2 Intelligent Essay Assessor

Intelligent essay assessor (Foltz et al., 1999) uses LSA for automatic scoring of short essays that would be used in any kind of content-based courses. Student essays are characterized by LSA representations of the meaning of their contained words and compared with pre-graded essays on degree of conceptual relevance and amount of relevant content by means of two kinds of scores: (1) the *holistic score*, the score of the closest pre-graded essay and (2) the *gold standard*, the LSA proximity between the student essay and a standard essay.

2.2.3 Summary Street

Summary Street (Kintsch et al., 2000) is also built on top of LSA. It helps students to write good summaries. First of all, a student is provided with a general advice on how to write a summary, then the student selects a topic, reads the text and writes out a summary. LSA procedures are then applied to give a holistic grade to the summary.

2.2.4 Apex

Apex (Dessus et al., 2000) is a web-based learning environment which manages student productions, assessments and courses. Once connected to the system, a student selects a topic or a question that he or she wishes to work on. The student then types a text about this topic into a text editor. At any time, she can get a three-part evaluation of the essay based on content, outline and coherence. At the content level, the system identifies how well the notions are covered by requesting LSA to measure a semantic similarity between the student text and each notion of the selected topic and correspondingly provides a message to the student.

3 Syntactically Enhanced LSA (SELSA)

LSA is based on word-document co-occurrence, also called a 'bag-of-words' approach. It is therefore blind to word-order or syntactic information. This puts limitations on LSA's ability to capture the meaning of a sentence which depends upon both syntax and semantics. The syntactic information in a text can be characterized in various ways like a full parse tree, a shallow parse, POS tag sequence etc. In an effort to generalize the LSA, we present here a concept of word-tag-document structure, which captures the behavior of a word within each syntactic context across various semantic contexts. The idea behind this is that the syntactic-semantic sense of a word is specified by the syntactic neighborhood in which it occurs. So representation of each such variation in an LSA-like space gives us a finer resolution in a word's behavior compared to an average behavior captured by LSA. This then allows to compare two text documents based on their syntactic-semantic regularity and not based on semantics-only. So it can be used in high quality text evaluation applications.

This approach is quite similar to the *tagged LSA* (Wiemer-Hastings and Zipitria, 2001) which considered a word along with its POS tag to discriminate multiple syntactic senses of a word. But our approach is an extension of this work towards a more general framework where a word along with the syntactic context specified by its adjacent words is considered as a unit of knowledge representation. We define the syntactic context as the POS tag information around a focus word. In particular, we look at the POS tag of the preceding word also called *prevtag* for convenience. The motivation for this comes from statistical language modeling and left-to-right parsing literature where a word is predicted or tagged using its preceding words and their POS tags. Moreover, *prevtag* is used as an approximation to the notion of a *preceding parse* tree characterizing the word sequence before the focus word. But in general, we can also use the syntactic information from the words following the current word, e.g. *posttag*, the POS tag of the next word. However, one of the concerns while incorporating syntactic information in LSA is that of sparse data estimation problem. So it is very important to choose a robust characterization of syntactic neighbourhood as well as apply smoothing either at the matrix formation level or at the time of projecting a document in the latent space.

The approach consists of first identifying a sufficiently large corpus representing the domain of tutoring. Then a POS tagger is used to convert it to a POS tagged corpus. The next step is to construct a matrix whose rows correspond to *word-prevtag* pairs and columns correspond to documents in the corpus. Again, a document can be a sentence, a paragraph or a larger unit of text. If the vocabulary size is I, POS tag vocabulary size is J and number of documents in corpus is K, then the matrix will be $IJ \times K$. Let $c_{i\_j,k}$ denote the frequency of word w_i with prevtag p_j in the document d_k. The notation i_j (i *underscore* j) in subscript is used for convenience and indicates word w_i with prevtag p_j i.e., $(i-1)J + j$th row of the matrix. Then as in LSA (Bellegarda, 2000), we find entropy ε_{i_j} of each *word-prevtag* pair and scale the corresponding row of the matrix by $(1 - \varepsilon_{i\_j})$. The document length normalization to each column of the matrix is also applied by dividing the entries of kth document by n_k, the number of words in document d_k. Let t_{i_j} be the frequency of i_jth *word-prevtag* pair in the whole corpus i.e. $t_{i\_j} = \sum_{k=1}^{K} c_{i\_j,k}$. Then ε_{i_j} and the matrix element $x_{i\_j,k}$ are given as:

$$\varepsilon_{i\_j} = -\frac{1}{\log K} \sum_{k=1}^{K} \frac{c_{i\_j,k}}{t_{i\_j}} \log \frac{c_{i\_j,k}}{t_{i\_j}} \qquad (1)$$

$$x_{i\_j,k} = (1 - \varepsilon_{i\_j}) \frac{c_{i\_j,k}}{n_k} \quad (2)$$

Once the matrix $\mathbf{X}$ is obtained, we perform its singular value decomposition (SVD) and approximate it by keeping the largest R singular values and setting the rest to zero. Thus,

$$\mathbf{X} \approx \hat{\mathbf{X}} = \mathbf{USV^T} \quad (3)$$

where, $\mathbf{U}(IJ \times R)$ and $\mathbf{V}(K \times R)$ are orthonormal matrices and $\mathbf{S}(R \times R)$ is a diagonal matrix. It is this dimensionality reduction step through SVD that captures major structural associations between *words-prevtag*s and documents, removes 'noisy' observations and allows the same dimensional representation of *words-prevtag*s and documents (albeit, in different bases). This R-dimensional space can be called either *syntactically enhanced latent semantic space* or *latent syntactic-semantic space*.

After the knowledge is represented in the latent syntactic-semantic space, we can project any new document as a R dimensional vector $\hat{\mathbf{d}}_L$ in this space. Let $\mathbf{d}$ be the $IJ \times 1$ vector representing this document whose elements d_{i_j} are the frequency counts i.e. number of times word w_i occurs with *prevtag* p_j, weighted by its corresponding entropy measure $(1 - \varepsilon_{i\_j})$. It can be thought of as an additional column in the matrix $\mathbf{X}$, and therefore can be thought of as having its corresponding vector $\mathbf{v}$ in the matrix $\mathbf{V}$. Then, $\mathbf{d} = \mathbf{USv^T}$ and

$$\hat{\mathbf{d}}_L = \mathbf{Sv^T} = \mathbf{U^T d} \quad (4)$$

which is a $R \times 1$ dimensional vector representation of the document in the latent space.

We can also define a syntactic-semantic similarity measure between any two text documents as the cosine of the angle between their projection vectors in the latent syntactic-semantic space. With this measure we can address the problems that LSA has been applied to, namely natural language understanding, cognitive modeling, statistical language modeling etc.

4 Experiment - Evaluating Students' Answers

We have studied the performance of SELSA and compared it with LSA in the AutoTutor task (section 2.2.1) for natural language understanding and cognitive modeling performance. The details of the experiment are presented below.

4.1 Corpus

The tutoring research group at the University of Memphis has developed the training as well as testing corpus for the AutoTutor task. The training corpus consisted of two complete computer literacy textbooks, and ten articles on each of the tutoring topics *viz.* hardware, operating system and the Internet. The test corpus was formed in the following manner : eight questions from each of the three topics were asked to a number of students. Then eight answers per question, 192 in total, were selected as test database. There were also around 20 good answers per question which were used in training and testing. Using this corpus, we have implemented LSA and SELSA.

4.2 Human Evaluation of Answers

For comparing the performance of SELSA and LSA with humans, we selected four human evaluators from computer related areas. Three of them were doctorate candidates and one had completed it, thus they were expert human evaluators. Each of them were given the 192 student-answers and a set of good answers to each of the question. They were asked to evaluate the answers on the basis of *compatibility score* i.e. the fraction of the number of sentences in a student-answer that matches any of the good answers. Thus, the score for each answer ranged between 0 to 1. They were not told what constitutes a "match", but were to decide themselves.

4.3 Syntactic Information

We approximated the syntactic neighborhood by the POS tag of preceding word. POS tagging was performed by the LTPOS software from the Language Technology Group of University of Edinburgh[1]. We also mapped the 45 tags from Penn tree-bank tagset to 12 tags so as to consider major syntactic categories and also to keep the size of resulting matrix manageable.

4.4 LSA and SELSA Training

We considered a paragraph as a unit of document. After removing very small documents consisting less than four words, we had 5596 documents. The vocabulary size, after removing words with frequency less than two and some stopwords, was 9194. The density of LSA and SELSA matrices were 0.27% and 0.025% respectively. SVD was performed using the MATLAB sparse matrix toolbox. We performed SVD with dimensions R varying from 200 to 400 in steps of 50.

4.5 Evaluation Measure

In order to evaluate the performance of SELSA and LSA on AutoTutor task, we need to define an appropriate measure. The earlier studies on this task used a correlation coefficient measure between the LSA's rating and human rating of the 192 answers. We have also used this as one of the three measures for comparison. But for a task having small sample size, the correlation coefficient is not reliably estimated, so we defined two new performance

[1] http://www.ltg.ed.ac.uk

measures. The first one was the mean absolute difference between the human and SELSA (correspondingly LSA) evaluations. In the other measure we used the comparison of how many answers were correctly evaluated versus how many were falsely evaluated by SELSA (LSA) as compared to human evaluations. A detailed explanation of these measures is given in the following section.

5 Results and Discussions

We calculated the compatibility score evaluation using SELSA (LSA) in an analogous way to the human evaluation. Thus SELSA (LSA) would evaluate the answers in the following manner. It would first break each student-answer into a number of sentences and then evaluate each sentence against the good answers for that question. If the cosine measure between the SELSA (LSA) representation of the sentence and any good answer exceeded a predefined threshold then that part was considered correct. Thus it would find the fraction of the number of sentences in a student-answer that exceeded the threshold. We performed the experiments by varying threshold between 0.05 to 0.95 with a step of 0.05. We also varied the number of singular values R from 200 to 400 with a step of 50. In the following, we present our results using the three evaluation measures.

5.1 Correlation Analysis

For each of the five SVD dimensions R and each value of the thresholds, we calculated the correlation coefficient between the SELSA (LSA) evaluation and each human rater's evaluation. Then we averaged this across the four human evaluators. The resulting average correlation curves for SELSA and LSA are shown in figs. (1) and (2) respectively.

From these two figures we observe that maximum correlation between SELSA and human raters is 0.47 and that between LSA and human is 0.51 while the average inter-human correlation was 0.59. Thus LSA seems to be closer to human than SELSA in this particular tutoring task. This seems to support the arguments from (Landauer et al., 1997) that syntax plays little role, if any, in semantic similarity judgments and text comprehension. But the likely reason behind this could be that the corpus, particularly the student answers, contained very poor syntactic structure and also that human evaluators might not have paid attention to grammatical inaccuracies in this technical domain of computer literacy.

But it is also worth noting that SELSA is closer to LSA than a previous approach of adding syntactic information to LSA (Wiemer-Hastings, 2000), which had a correlation of 0.40 compared to 0.49 of LSA on the same task of evaluating students' answers, where average inter-human correlation was 0.78 between the expert raters and 0.51 between the intermediate experts. SELSA

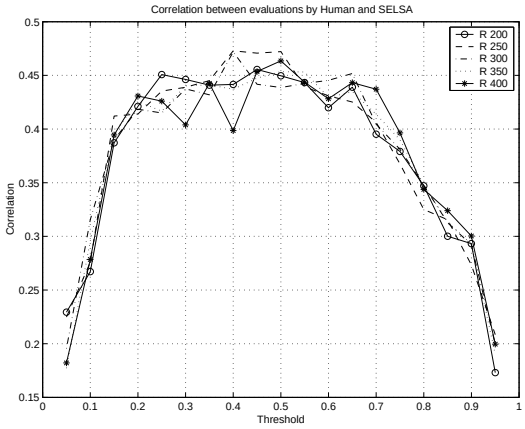

Figure 1: Correlation between SELSA and human evaluators

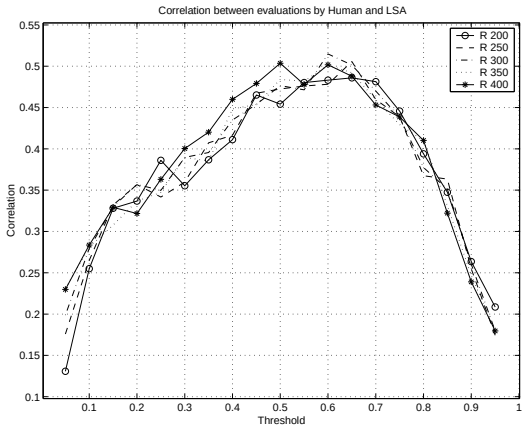

Figure 2: Correlation between LSA and human evaluators

is also comparable to *tagged LSA* (Wiemer-Hastings and Zipitria, 2001), which used the current POS tag instead of *prevtag*. It had a correlation of 0.27 compared to 0.36 of LSA in a modified evaluation task of judging similarity between two sentences where the correlation between skilled raters was 0.45 and that between non-proficient raters was 0.35.

If we look at these curves more carefully, especially, their behavior across thresholds, then it is interesting to note that SELSA has wider threshold-widths(TW) than LSA across all the cases of SVD dimension R. In table (1) and (2) we have shown the 10% and 20% TW of SELSA and LSA respectively. This is calculated by finding the range over thresholds for which the correlation is within 10% and 20% of the maximum correlation. This observation shows that SELSA is much more robust across thresholds than LSA in the sense that semantic information is discriminated better in SELSA space than in LSA space.

R	Cor_{max}	T_{max}	10% TW	20% TW
200	0.46	0.45	0.48	0.63
250	0.47	0.40	0.42	0.61
300	0.47	0.40	0.41	0.62
350	0.45	0.50	0.55	0.65
400	0.46	0.50	0.45	0.64

Table 1: Threshold Width of SELSA

R	Cor_{max}	T_{max}	10% TW	20% TW
200	0.49	0.65	0.33	0.44
250	0.51	0.65	0.29	0.44
300	0.51	0.60	0.26	0.41
350	0.50	0.60	0.32	0.44
400	0.50	0.50	0.32	0.50

Table 2: Threshold Width of LSA

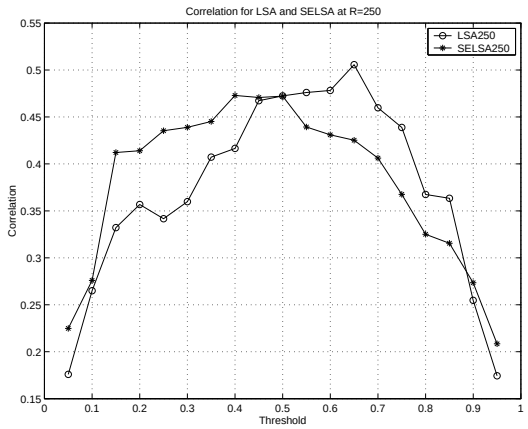

Figure 3: LSA vs SELSA for SVD dimensions 250

Another interesting observation occurs when we plot the two curves simultaneously as shown in fig. (3). Here we plotted the SELSA and LSA performances for 250 dimensions of latent space. We can easily see that SELSA performs better than LSA for thresholds less than 0.5 and viceversa. This observation along with the previous observation about TW can be understood in the following manner. When comparing two document vectors for a cosine measure exceeding a threshold, we can consider one of the vectors to be the axis of a right circular cone with a semi-vertical angle decided by the threshold. If the other vector falls within this cone, we say the two documents are matching. Now if the human raters emphasized semantic similarity, which is most likely the case, then this means that LSA could best capture the same information in a narrower cone while SELSA required a wider cone. This is quite intuitive in the sense that SELSA has zoomed the document similarity measure axis by putting finer resolution of syntactic information. Thus mere semantically similar documents are placed wider apart in SELSA space than syntactic-semantically similar documents. This concept can be best used in a language modeling task where a word is to be predicted from the history. It is observed in (Kanejiya et al., 2003) that SELSA assigns better probabilities to syntactic-semantically regular words than LSA, although the overall perplexity reduction over a bi-gram language model was less than that by LSA.

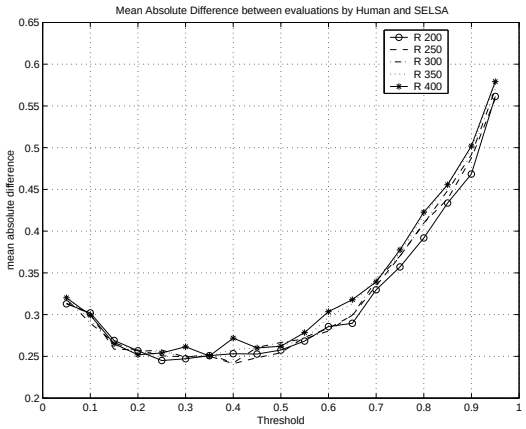

Figure 4: Mean absolute difference between SELSA and human evaluators

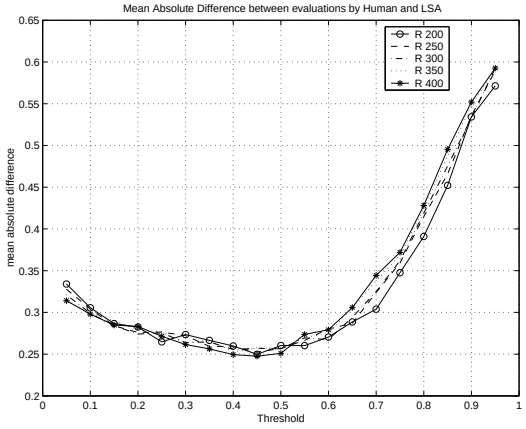

Figure 5: Mean absolute difference between LSA and human evaluators

5.2 Mean Absolute Difference Analysis

Here we calculated the mean absolute difference(*MAD*) between a human rater's evaluation and SELSA (LSA) evaluations as follow:

$$MAD = \frac{1}{192} \sum_{i=1}^{192} |h_i - l_i| \qquad (5)$$

where, h_i and l_i correspond to human and SELSA(LSA) evaluation of i^{th} answer. This was then averaged across human evaluators. These results are plotted in figs. (4) and (5). These two curves show that SELSA and LSA are almost equal to each other. Again SELSA has the advantage of more robustness and in most cases it is even better than LSA in terms of minimum *MAD* with human. Tables (3) and (4) show values of minimum *MAD* at various values of SVD dimensions R. The best minimum *MAD* for SELSA is 0.2412 at 250 dimensional space while that for LSA is 0.2475 at 400 dimensions. The average *MAD* among human evaluators is 0.2050.

R	$minMAD$	$maxCorrect$	$minFalse$
200	0.2449	125	31
250	0.2412	125	30
300	0.2422	126	30
350	0.2484	125	31
400	0.2504	124	32

Table 3: SELSA - MAD, correct and false evaluation

R	$minMAD$	$maxCorrect$	$minFalse$
200	0.2497	122	29
250	0.2523	120	31
300	0.2555	121	32
350	0.2525	122	32
400	0.2475	123	30

Table 4: LSA - MAD, correct and false evaluation

5.3 Correct vs False Evaluations Analysis

We define an evaluation l_i by SELSA (LSA) to be correct or false as below:

$l_i \quad CORRECT \quad$ if $|l_i - h_i| < CT$
$l_i \quad FALSE \quad$ if $|l_i - h_i| > FT$

where CT and FT are correctness and falsehood thresholds which were set to 0.05 and 0.95 respectively for strict measures. Number of such correct as well as false evaluations were then averaged across the four human evaluators. They are plotted in figs. (6) and (7) for SELSA and LSA respectively (the upper curves corresponding to correct and the lower ones to false evaluations). The maximum number of correct ($maxCorrect$) and the minimum number of false ($minFalse$) evaluations across the thresholds for each value of SVD dimensions are calculated and shown in tables (3) and (4). We observe that the best performance for SELSA is achieved at 300 dimensions with 126 correct and 30 false evaluations, while for LSA it is at 400 dimensions with 123 correct and 30 false evaluations. The average correct and false evaluations among all human-human evaluator pairs were 132 and 23 respectively. Thus here also SELSA is closer to human evaluators than LSA. In fact, for the cognitive task like AutoTutor, this is a more appealing and explicit measure than the previous two. Apart from these three measures, one can also calculate precision, recall and F-measure (Burstein et al., 2003) to evaluate the performance.

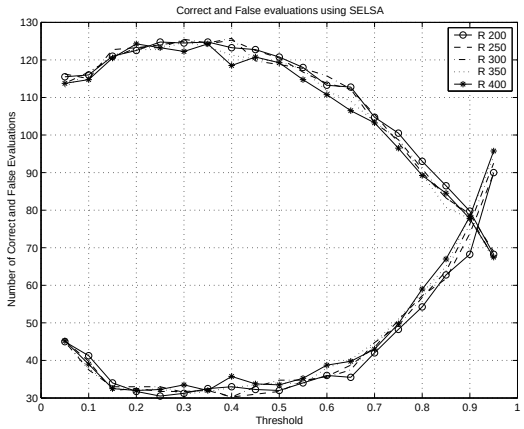

Figure 6: Correct and false evaluations by SELSA as compared to human evaluators

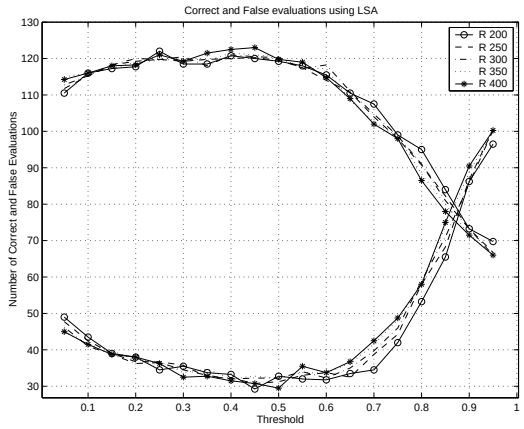

Figure 7: Correct and False evaluations by LSA as compared to human evaluators

6 Conclusion

Automatic evaluation of students' answers in an intelligent tutoring system can be performed using LSA. But LSA lacks syntactic information which can be also useful for meaning representation of a text document. So, we have developed and implemented a model called syntactically enhanced LSA which generalizes LSA by augmenting a word with the POS tag of the preceding word to derive a latent syntactic-semantic information. Experimental results on the AutoTutor task of evaluating students' answers to computer science questions show a range of

performance comparison between SELSA and LSA. In terms of the correlation measure with human raters, LSA is slightly better than SELSA. But SELSA is at least as good as LSA in terms of the mean absolute difference measure. On the other end, SELSA is able to correctly evaluate a few more answers than LSA is. SELSA can do better if the training and testing corpora have a good syntactic structure.

From the correlation performance analysis, it is observed that SELSA is more robust in discriminating the semantic information across a wider threshold width than LSA. It is also found that SELSA uses the syntactic information to expand the document similarity measure i.e., mere semantically similar documents are placed wider apart than syntactic-semantically similar documents in SELSA space.

These initial results are part of an ongoing research towards an overall improvement of natural language understanding and modeling. Although the present version of SELSA has limited improvements over LSA, it leads to future experiments with robust characterization of syntactic neighbourhood in terms of headwords or phrase structure as well as applying smoothing across syntax to tackle the problem of sparse data estimation.

References

V. Aleven, O. Popescu, and K. R. Koedinger. 2001. A tutorial dialogue system with knowledge-based understanding and classification of student explanations. In *Working notes of the 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Seattle.

J. R. Bellegarda. 2000. Exploiting latent semantic information in statistical language modeling. *Proceedings of the IEEE*, 88(8):1279–1296.

J. Burstein, C. Leacock, and M. Chodorow. 2003. Criterion: Online essay evaluation: An application for automated evaluation of student essays. In *Proc. of the Fifteenth Annual Conf. on Innovative Applications of Artificial Intelligence*, Acapulco, Mexico. (in press).

S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshmann. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

P. Dessus, B. Lemaire, and A. Vernier. 2000. Free-text assessment in a virtual campus. In *Proc. 3rd Int. Conf. on Human Systems Learning (CAPS'2000)*, Paris.

P. W. Foltz, D. Laham, and T. K. Landauer. 1999. Automated essay scoring: applications to educational technology. In *Proc. of the ED-MEDIA'99 conference*, Charlottesville. AACE.

R. Freedman, C. P. Ros, M. A. Ringenberg, and K. VanLehn. 2000. ITS tools for natural language dialogue: A domain-independent parser and planner. In *Fifth International Conference on Intelligent Tutoring Systems (ITS 2000)*, Montreal. Springer-Verlag.

M. Glass. 2001. Processing language input in the CIRCSIM-tutor intelligent tutoring system. In J. D. Moore, C. L. Redfield, and W. L. Johnson, editors, *Artificial Intelligence in Education*, pages 210–221. IOS Press, San Antonio.

A. C. Graesser, K. Wiemer-Hastings, P. Wiemer-Hastings, R. Kreuz, and Tutoring Research Group. 1999. Autotutor: A simulation of a human tutor. *Journal of Cognitive Systems Research*, 1:35–51.

A. C. Graesser, P. Wiemer-Hastings, K. Wiemer-Hastings, D. Harter, Tutoring Research Group, and N. Person. 2000. Using latent semantic analysis to evaluate the contributions of students in autotutor. *Interactive Learning Environments*, 8(2):129–147.

D. P. Kanejiya, A. Kumar, and S. Prasad. 2003. Statistical language modeling using syntactically enhanced LSA. In *Proc. TIFR Workshop on Spoken Language Processing*, pages 93–100, Mumbai, India.

E. Kintsch, D. Steinhart, G. Stahl, and the LSA Research Group. 2000. Developing summarization skills through the use of lsa-based feedback. *Interactive Learning Environments*, 8(2):87–109.

T. K. Landauer, D. Laham, B. Rehder, and M. E. Schreiner. 1997. How well can passage meaning be derived without using word order? a comparison of latent semantic analysis and humans. In *Proc. 9th annual meeting of the Cognitive Science Society*, pages 412–417, Mawhwah, NJ. Erlbaum.

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

B. Lemaire. 1999. Tutoring systems based on latent semantic analysis. In S. Lajoie and M. Vivet, editors, *Artificial Intelligence in Education*, pages 527–537. IOS Press, Amsterdam.

P. Wiemer-Hastings and I. Zipitria. 2001. Rules for syntax, vectors for semantics. In *Proc. 23rd Annual Conf. of the Cognitive Science Society*, Mawhwah, NJ. Erlbaum.

P. Wiemer-Hastings, A. C. Graesser, D. Harter, and Tutoring Research Group. 1998. The foundation and architecture of autotutor. In *Proc. 4th Int. Conf. on Intelligent Tutoring Systems*, Berlin. Springer-Verlag.

P. Wiemer-Hastings. 2000. Adding syntactic information to lsa. In *Proc. 22nd Annual Conf. of the Cognitive Science Society*, pages 988–993, Mawhwah, NJ. Erlbaum.