Statistical Language Modeling with Performance Benchmarks using Various Levels of Syntactic-Semantic Information

Dharmendra KANEJIYA[*], **Arun KUMAR**[†], **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

Statistical language models using n-gram approach have been under the criticism of neglecting large-span syntactic-semantic information that influences the choice of the next word in a language. One of the approaches that helped recently is the use of latent semantic analysis to capture the semantic fabric of the document and enhance the n-gram model. Similarly there have been some approaches that used syntactic analysis to enhance the n-gram models. In this paper, we explain a framework called syntactically enhanced latent semantic analysis and its application in statistical language modeling. This approach augments each word with its syntactic descriptor in terms of the part-of-speech tag, phrase type or the supertag. We observe that given this syntactic knowledge, the model outperforms LSA based models significantly in terms of perplexity measure. We also present some observations on the effect of the knowledge of content or function word type in language modeling. This paper also poses the problem of better syntax prediction to achieve the benchmarks.

1 Introduction

Statistical language models consist of estimating the probability distributions of a word given the history of words so far used. The standard n-gram language model considers two histories to be equivalent if they end in the same $n-1$ words. Due to the tradeoff between predictive power and reliability of estimation, n is typically chosen to be 2 (bi-gram) or 3 (tri-gram). Even tri-gram model suffers from sparse-data estimation problem, but various smoothing techniques (Goodman, 2001) have led to significant improvements in many applications. But still the criticism that n-grams are unable to capture the long distance dependencies that exist in a language, remains largely valid.

In order to model the linguistic structure that spans a whole sentence or a paragraph or even more, various approaches have been taken recently. These can be categorized into two main types : *syntactically motivated* and *semantically motivated* large span consideration. In the first type, probability of a word is decided based on a parse-tree information like grammatical headwords in a sentence (Charniak, 2001) (Chelba and Jelinek, 1998), or based on part-of-speech (POS) tag information (Galescu and Ringger, 1999). Examples of the second type are (Bellegarda, 2000) (Coccaro and Jurafsky, 1998), where *latent semantic analysis* (LSA) (Landauer et al., 1998) is used to derive large-span semantic dependencies. LSA uses word-document co-occurrence statistics and a matrix factorization technique called singular value decomposition to derive semantic similarity measure between any two text units - words or documents. Each of these approaches, when integrated with n-gram language model, has led to improved performance in terms of perplexity as well as speech recognition accuracy.

While each of these approaches has been studied independently, it would be interesting to see how they can be integrated in a unified framework which looks at syntactic as well as semantic information in the large span. Towards this direction, we describe in this paper a mathematical framework called *syntactically enhanced latent syntactic-semantic analysis* (SELSA). The basic hypothesis is that by considering a word alongwith its syntactic descriptor as a unit of knowledge representation in the LSA-like framework, gives us an approach to joint syntactic-semantic analysis of a document. It also provides a finer resolution in each word's semantic description for each of the syntactic contexts it occurs in. Here the syntactic descriptor can come from various levels e.g. part-of-speech tag, phrase type, supertag etc. This syntactic-semantic representation can be used in language modeling to allocate the probability mass to words in accordance with their semantic similarity to the history as well as syntactic fitness to the local context.

In the next section, we present the mathematical framework. Then we describe its application to statistical language modeling. In section 4 we explain the the use of various levels of syntactic information in SELSA. That is followed by experimental results and conclusion.

2 Syntactically Enhanced LSA

Latent semantic analysis (LSA) is a statistical, algebraic technique for extracting and inferring relations of expected contextual usage of words in documents (Landauer et al., 1998). It is based on word-document co-occurrence statistics, and thus is often called a 'bag-of-words' approach. It neglects the word-order or syntactic information in a language which if properly incorporated, can lead to better language modeling. In an effort to include syntactic information in the LSA framework, we have developed a model which characterizes a word's behavior across various syntactic and semantic contexts. This can be achieved by augmenting a word with its syntactic descriptor and considering as a unit of knowledge representation. The resultant LSA-like analysis is termed as *syntactically enhanced latent semantic analysis (SELSA)*. This approach can better model the finer resolution in a word's usage compared to an average representation by LSA. This finer resolution can be used to better discriminate semantically ambiguous sentences for cognitive modeling as well as to predict a word using syntactic-semantic history for language modeling. We explain below, a step-by-step procedure for building this model.

2.1 Word-Tag-Document Structure

The syntactic description of a word can be in many forms like part-of-speech tag, phrase type or supertags. In the description hereafter we call any such syntactic information as tag of the word. Now, consider a tagged training corpus of sufficient size in the domain of interest. The first step is to construct a matrix whose rows correspond to *word-tag* pairs and columns correspond to documents in the corpus. A document can be a sentence, a paragraph or a larger unit of text. If the vocabulary $\mathcal{V}$ consists of I words, tagset $\mathcal{T}$ consists of J tags and the 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 *tag* t_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 tag t_j i.e., $(i-1)J+j$th row of the matrix. Then we find entropy ε_{i_j} of each *word-tag* 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 c_{i_j} be the frequency of i_jth *word-tag* pair in the whole corpus i.e. $c_{i\_j} = \sum_{k=1}^{K} c_{i\_j,k}$. Then,

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

$$\varepsilon_{i\_j} = -\frac{1}{\log K}\sum_{k=1}^{K}\frac{c_{i\_j,k}}{c_{i\_j}}\log\frac{c_{i\_j,k}}{c_{i\_j}} \qquad (2)$$

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,

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

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-tag*s and documents, removes 'noisy' observations and allows the same dimensional representation of *words-tag*s and documents (albeit, in different bases).This same dimensional representation is used (eq. 12) to find syntactic-semantic correlation between the present word and the history of words and then to derive the language model probabilities. This R-dimensional space can be called either *syntactically enhanced latent semantic space* or *latent syntactic-semantic space.*

2.2 Document Projection in SELSA Space

After the knowledge is represented in the latent syntactic-semantic space, we can project any new document as a R dimensional vector $\bar{\mathbf{v}}_{selsa}$ in this space. Let the new document consist of a word sequence $w_{i_1}, w_{i_2}, \ldots, w_{i_n}$ and let the corresponding tag sequence be $t_{j_1}, t_{j_2}, \ldots, t_{j_n}$, where i_p and j_p are the indices of the pth word and its tag in the vocabulary $\mathcal{V}$ and the tagset $\mathcal{T}$ respectively. 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 *tag* 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

$$\bar{\mathbf{v}}_{selsa} = \mathbf{vS} = \mathbf{d^T U} = \frac{1}{n}\sum_{p=1}^{n}(1 - \varepsilon_{i_p\_j_p})\mathbf{u}_{i_p\_j_p} \quad (4)$$

which is a $1 \times R$ dimensional vector representation of the document in the latent space. Here $\mathbf{u}_{i_p\_j_p}$ represents the row vector of the SELSA $\mathbf{U}$ matrix corresponding to the word w_{i_p} and tag t_{j_p} in the current document.

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.

3 Statistical Language Modeling using SELSA

3.1 Framework

We follow the framework in (Bangalore, 1996) to define a class-based language model where classes are defined by the tags. Here probability of a sequence W_n of n words is given by

$$\begin{aligned} P(W_n) &= \sum_{t_1}\cdots\sum_{t_n}\prod_{q=1}^{n} P(w_q|t_q, W_{q-1}, T_{q-1}) \\ & \quad P(t_q|W_{q-1}, T_{q-1}) \end{aligned} \quad (5)$$

where t_i is a tag variable for the word w_i. To compute this probability in realtime based on local information, we make certain assumptions:

$$\begin{aligned} P(w_q|t_q, W_{q-1}, T_{q-1}) &\approx P(w_q|t_q, w_{q-1}, w_{q-2}) \\ P(t_q|W_{q-1}, T_{q-1}) &\approx P(t_q|t_{q-1}) \end{aligned} \quad (6)$$

where probability of a word is calculated by renormalizing the tri-gram probability to those words which are compatible with the tag in context. Similarly, tag probability is modeled using a bi-gram model. Other models like tag based likelihood probability of a word or tag tri-grams can also be used. Similarly there is a motivation for using the syntactically enhanced latent semantic analysis method to derive the word probability given the syntax of tag and semantics of word-history.

The calculation of perplexity is based on conditional probability of a word given the word history, which can be derived in the following manner using recursive computation.

$$\begin{aligned} & P(w_q|W_{q-1}) \\ &= \sum_{t_q} P(w_q|t_q, W_{q-1}) P(t_q|W_{q-1}) \\ &\approx \sum_{t_q} P(w_q|t_q, w_{q-1}, w_{q-2}) \sum_{t_{q-1}} \\ & \quad P(t_q|t_{q-1}) P(t_{q-1}|W_{q-1}) \\ &= \sum_{t_q} P(w_q|t_q, w_{q-1}, w_{q-2}) \sum_{t_{q-1}} \\ & \quad P(t_q|t_{q-1}) \frac{P(W_{q-1}, t_{q-1})}{\sum_{t_{q-1}} P(W_{q-1}, t_{q-1})} \end{aligned} \quad (7)$$

where,

$$\begin{aligned} & P(W_q, t_q) \\ &= \left(\sum_{t_{q-1}} P(W_{q-1}, t_{q-1}) P(t_q|t_{q-1})\right) \\ & \quad P(w_q|t_q, w_{q-1}, w_{q-2}) \end{aligned} \quad (8)$$

A further reduction in computation is achieved by restricting the summation over only those tags which the target word can anchor. A similar expression using the tag tri-gram model can be derived which includes double summation. The efficiency of this model depends upon the prediction of tag t_q using the word history W_{q-1}. When the target tag is correctly known, we can derive a performance benchmark in terms of lower bound on the perplexity achievable. Furthermore, if we assume tagged corpus, then t_q's and T_q's become deterministic variables and (5) and (7) can be written as,

$$P(W_n) = \prod_{q=1}^{n} P(w_q|t_q, W_{q-1}, T_{q-1}) \quad (9)$$

$$P(w_q|W_{q-1}) = P(w_q|t_q, W_{q-1}, T_{q-1}) \quad (10)$$

respectively in which case the next described SELSA language model can be easily applied to calculate the benchmarks.

3.2 SELSA Language Model

SELSA model using tag information for each word can also be developed and used along the line of LSA based language model. We can observe in the above framework the need for the probability of the form $P(w_q|t_q, W_{q-1}, T_{q-1})$ which can be evaluated using the SELSA representation of the *word-tag* pair corresponding to w_q and t_q and the history $W_{q-1}T_{q-1}$. The former is given by the row $\mathbf{u}_{i_q\text{-}j_q}$ of SELSA $\mathbf{U}$ matrix and the later can be projected onto the SELSA space as a vector $\tilde{\bar{\mathbf{v}}}_{q-1}$ using (4). The length of history can be tapered to reduce the effect of far distant words using the exponential forgetting factor $0 < \lambda < 1$ as below:

$$\tilde{\bar{\mathbf{v}}}_{q-1} = \frac{1}{q-1} \sum_{p=1}^{q-1} \lambda^{q-1-p} (1 - \varepsilon_{i_p\text{-}j_p}) \mathbf{u}_{i_p\text{-}j_p} \quad (11)$$

The next step is to calculate the cosine measure reflecting the syntactic-semantic 'closeness' between the word w_q and the history W_{q-1} as below:

$$K(w_q, W_{q-1}) = \frac{\mathbf{u}_{i_q\text{-}j_q} \tilde{\bar{\mathbf{v}}}_{q-1}^T}{\| \mathbf{u}_{i_q\text{-}j_q} \mathbf{S}^{\frac{1}{2}} \| \| \tilde{\bar{\mathbf{v}}}_{q-1} \mathbf{S}^{-\frac{1}{2}} \|} \quad (12)$$

Then SELSA based probability $P^{(sel)}(w_q|W_{q-1})$ is calculated by allocating total probability mass in proportion to this closeness measure such that least likely word has a probability of 0 and all probabilities sum to 1:

$$K_{min}(W_{q-1}) = \min_{w_i \in \mathcal{V}} K(w_i, W_{q-1}) \quad (13)$$

$$\hat{P}(w_q|W_{q-1}) = \frac{K(w_q, W_{q-1}) - K_{min}(W_{q-1})}{\sum_{w_i \in \mathcal{V}} (K(w_i, W_{q-1}) - K_{min}(W_{q-1}))} \quad (14)$$

But this results in a very limited dynamic range for SELSA probabilities which leads to poor performance. This is alleviated by raising the above derived probability to a power $\gamma > 1$ and then normalizing as follows(Coccaro and Jurafsky, 1998):

$$P^{(sel)}(w_q|W_{q-1}) = \frac{\hat{P}(w_q|W_{q-1})^{\gamma}}{\sum_{w_i \in \mathcal{V}} \hat{P}(w_i|W_{q-1})^{\gamma}} \quad (15)$$

This probability gives more importance to the large span syntactic-semantic dependencies and thus would be higher for those words which are syntactic-semantically regular in the recent history as compared to others. But it will not predict very well certain locally regular words like *of, the* etc whose main role is to support the syntactic structure in a sentence. On the other hand, n-gram language models are able to model them well because of maximum likelihood estimation from training corpus and various smoothing techniques. So the best performance can be achieved by integrating the two. One way to derive the 'SELSA + N-gram' joint probability $P^{(sel+ng)}(w_q|W_{q-1})$ is to use the geometric mean based integration formula given for LSA in (Coccaro and Jurafsky, 1998) as follows:

$$P^{(sel+ng)}(w_q|W_{q-1}) = \frac{\left[P^{(sel)}(w_q|W_{q-1})\right]^{\xi_{i_q}} \left[P(w_q|w_{q-1},\ldots,w_{q-n+1})\right]^{1-\xi_{i_q}}}{\sum_{w_i \in \mathcal{V}} \left[P^{(sel)}(w_i|W_{q-1})\right]^{\xi_i} \left[P(w_i|w_{q-1},\ldots,w_{q-n+1})\right]^{1-\xi_i}} \quad (16)$$

where, $\xi_{i_q} = \frac{1-\varepsilon_{i_q\text{-}j_q}}{2}$ and $\xi_i = \frac{1-\varepsilon_{i\text{-}j_q}}{2}$ are the geometric mean weights for SELSA probabilities for the current word w_q and any word $w_i \in \mathcal{V}$ respectively.

4 Various Levels of Syntactic Information

In this section we explain various levels of syntactic information that can be incorporated within SELSA framework. They are supertags, phrase type and content/fuction word type. These are in decreasing order of complexity and provide finer to coarser levels of syntactic information.

4.1 Supertags

Supertags are the elementary structures of Lexicalized Tree Adjoining Grammars (LTAGs) (Bangalore and Joshi, 1999). They are combined by the operations of substitution and adjunction to yield a parse for the sentence. Each supertag is lexicalized i.e. associated with at least one lexical item - the anchor. Further, all the arguments of the anchor of a supertag are localized within the same supertag which allows the anchor to impose syntactic and semantic (predicate-argument) constraints directly on its arguments. As a result, a word is typically associated with one supertag for each syntactic configuration the word may appear in. Supertags can be seen as providing a much more refined set of classes than do part-of-speech tags and hence we expect supertag-based language models to be better than part-of-speech based language models.

4.2 Phrase-type

Words in a sentence are not just strung together as a sequence of parts of speech, but rather they are organized into *phrases*, grouping of words that are clumped as a unit. A sentence normally rewrites as a subject noun phrase (NP) and a verb phrase (VP) which are the major types of phrases apart from propositional phrases, adjective phrases etc (Manning and Schutze, 1999). Using the two major phrase types and the rest considered as other type, we constructed a model for SELSA. This model assigns each word three syntactic descriptions depending on its frequency of occurrence in each of three phrase types across a number of documents. This model captures the semantic behaviour of each word in each phrase type. Generally nouns accur in noun phrases and verbs occur in verb phrases while prepositions occur in the other type. So this framework brings in the finer syntactic resolution in each word's semantic description as compared to LSA based average description. This is particularly more important for certain words occurring as both noun and verb.

4.3 Content or Function Word Type

If a text corpus is analyzed by counting word frequencies, it is observed that there are certain words which occur with very high frequencies e.g. *the, and, a, to* etc. These words have a very important grammatical behaviour, but they do not convey much of the semantics. Thse words are called *function or stop words.* Similarly in a text corpus, there are certain words with frequencies in moderate to low range e.g. *car, wheel, road* etc. They each play an important role in deciding the semantics associated with the whole sentence or document. Thus they are known as *content words.* Generally a list of vocabulary consists of a few hundred function words and a few tens of thousands of content words. However, they span more or less the same frequency space of a corpora. So it is also essential to give them equal importance by treating them separately in a language modeling framework as they both convey some sort of orthogonal information - syntactic vs semantic. LSA is better at predicting topic bearing content words while parsing based models are better for function words. Even n-gram models are quite better at modeling function words, but they lack the large-span semantic that can be achieved by LSA. On the other hand, SELSA model is suitable for both types of words as it captures semantics of a word in a syntactic context.

We performed experiments with LSA and SELSA with various levels of syntactic information in both the situations - content words only vs content and function words together. In the former case, the function words are treated by n-gram model only.

5 Experiments and Discussion

A statistical language model is evaluated by how well it predicts some hitherto unseen text - *test data* - generated by the source to be modeled. A commonly used quality measure for a given model $\mathcal{M}$ is related to the entropy of the underlying source and is known as *perplexity*(PPL). Given a word sequence $w_1, w_2, \ldots, w_N$ to be used as a test corpus, the perplexity of a language model $\mathcal{M}$ is given by:

$$PPL = \exp\left(-\frac{1}{N}\sum_{q=1}^{N}\log P^{(\mathcal{M})}(w_q|W_{q-1})\right) \quad (17)$$

Perplexity also indicates the (geometric) average branching factor of the language according to the model $\mathcal{M}$ and thus indicates the difficulty of a speech recognition task(Jelinek, 1999). The lower the perplexity, the better the model; usually a reduction in perplexity translates into a reduction in *word error rate* of a speech recognition system.

We have implemented both the LSA and SELSA models using the BLLIP corpus[1] which consists of machine-parsed English new stories from the Wall Street Journal (WSJ) for the years 1987, 1988 and 1989. We used the supertagger (Bangalore and Joshi, 1999) to supertag each word in the corpus. This had a tagging acuracy of 92.2%. The training corpus consisted of about 40 million words from the WSJ 1987, 1988 and some portion of 1989. This consists of about 87000 documents related to news stories. The test corpus was a section of WSJ 1989 with around $300,000$ words. The baseline tri-gram model had a perplexity of 103.12 and bi-gram had 161.06. The vocabulary size for words was 20106 and for supertags was 449.

5.1 Perplexity Results

In the first experiment, we performed SELSA using supertag information for each word. The word-supertag vocabulary was about 60000. This resulted in a matrix of about 60000X87000 for which we performed SVD at various dimensions. Similarly we trained LSA matrix and performed its SVD. Then we used this knowledge to calculate language model probability and then

[1] Available from the Linguistic Data Consortium(LDC) `www.ldc.upenn.edu`

integrated with tri-gram probability using geometric interpolation method (Coccaro and Jurafsky, 1998). In the process, we had assumed the knowledge of the content/function word type for the next word being predicted. Furthermore, in this experiment, we had used only content words for LSA as well as SELSA representation, while the function words were treated by tri-gram model only. We also used the supertagged test corpus, thus we knew the supertag of the next word being predicted. These results thus sets benchmarks for content word based SELSA model. With these assumptions, we obtained the perplexity values as shown in Table 1.

SVD dimensions R	LSA+ tri-gram	SELSA+ tri-gram
tri-gram only	103.12	103.12
0 (uniform prob)	78.92	60.83
2	78.05	60.88
10	74.92	57.88
20	72.91	56.15
50	69.85	52.80
125	68.42	50.39
200	67.79	49.50
300	67.34	48.84

Table 1: Perplexity at different SVD dimensions with content/function word type knowledge assumed. For SELSA, these are benchmarks with correct supertag knowledge.

These benchmark results show that given the knowledge of the content or function word as well as the supertag of the word being predicted, SELSA model performs far better than the LSA model. This improvement in the performance is attributed to the finer level of syntactic information available now in the form of supertag. Thus given the supertag, the choice of the word becomes very limited and thus perplexity decreases. The decrease in perplexity across the SVD dimension shows that the SVD also plays an important role and thus for SELSA it is truely a latent syntactic-semantic analysis. Thus if we devise an algorithm to predict the supertag of the next word with a very high accuracy, then there is a gurantee of performance improvement by this model compared to LSA.

Our next experiment, was based on no knowledge of content or function word type of the next word. Thus the LSA and SELSA matrices had all the words in the vocabulary. We also kept the SVD dimensions for both SELSA and LSA to 125. The results are shown in Table 2. In this case, we observe that LSA achieves the perplexity of 88.20 compared to the baseline tri-gram 103.12. However this is more than LSA perplexity of 68.42 when the knowledge of content/function words was assumed. This relative increase is mainly due to poor modeling of function words in the LSA-space. However for SELSA, we can observe that its perplexity of 36.37 is less than 50.39 value in the case of knowledge about content/function words. This is again attributed to better modeling of syntactically regular function words in SELSA. This can be better understood from the observation that there were 305 function words compared to 19801 content words in the vocabulary spanning 19.8 and 20.3 million words respectively in the training corpus. Apart from this, there were $152, 145$ and 147 supertags anchoring function word only, content word only and both types of words respectively. Thus given a supertag belonging to function word specific supertags, the 'vocabulary' for the target word is reduced by orders of magnitude compared to the case for content word specific supertags. It is also worth observing that the 125-dimensional SVD case of SELSA is better than the 0-dimensional SVD or uniform SELSA case. Thus the SVD plays a role in deciphering the syntactic-semantically important dimensions of the information space.

Model	Perplexity
tri-gram only	103.12
LSA(125)+tri-gram	88.20
SELSA(125)+tri-gram	36.37
uniform-SELSA+tri-gram	41.79

Table 2: Perplexity without content/function word knowledge. For SELSA, these are benchmarks with correct supertag knowledge.

We also performed experiments using the phrase-type (NP, VP, others) knowledge and incorporated them within SELSA framework. The resultant model was also used to calculate perplexity values and the results on content/function type assumption set compares favourably with LSA by improving the performance. In another experiment we used the part-of-speech tag of the previous word (*prevtag*) within SELSA, but it couldn't improve against the plain LSA. These results shows that phrase level information is somewhat useful if it can be predicted correctly, but previous POS tags are not useful.

Model	Perplexity
tri-gram only	103.12
LSA(125)+tri-gram	68.42
phrase-SELSA(125)+tri-gram	64.78
prevtag-SELSA(125)+tri-gram	69.12

Table 3: Perplexity of phrase/prevtag based SELSA with the knowledge of content/function word type and the correct phrase/prevtag

Finally the utility of this language model can be tested in a speech recognition experiment. Here it can be most suitably applied in a second-pass rescoring framework where the output of first-pass could be the N-best list of either joint word-tag sequences (Wang and Harper, 2002) or word sequences which are then passed through a syntax tagger. Both these approaches allow a direct application of the results shown in above experiments, however there is a possibility of error propagation if some word is incorrectly tagged. The other approach is to predict the tag left-to-right from the word-tag partial prefix followed by word prediction and then repeating the procedure for the next word.

6 Conclusions and Research Direction

We presented the effect of incorporating various levels of syntactic information in a statistical language model that uses the mathematical framework called syntactically enhanced LSA. SELSA is an attempt to develop a unified framework where syntactic and semantic dependencies can be jointly represented. It generalizes the LSA framework by incorporating various levels of the syntactic information along with the current word. This provides a mechanism for statistical language modeling where the probability of a word given the semantics of the preceding words is constrained by the adjacent syntax. The results on WSJ corpus sets a set of benchmarks for the performance improvements possible with these types of syntactic information. The supertag based information is very fine-grained and thus leads to a large reduction in perplexity if correct supertag is known. It is also observed that the knowledge of the phrase type also helps to reduce the perplexity compared to LSA. Even the knowledge of the content/function word type helps additionally in each of the SELSA based language models. These benchmarks can be approached with better algorithms for predicting the necessary syntactic information. Our experiments are still continuing in this direction as well as toward better understanding of the overall statistical language modeling problem with applications to speech recognition.

References

S. Bangalore and A. K. Joshi. 1999. Supertagging:an approach to almost parsing. *Computational Linguistics*, 25(2):237–265.

S. Bangalore. 1996. "almost parsing" technique for language modeling. In *Proc. Int. Conf. Spoken Language Processing*, Philadeplphia, PA, USA.

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

E. Charniak. 2001. Immediate-head parsing for language models. In *Proc. 39th Annual Meeting of the Association for Computational Linguistics.*

C. Chelba and F. Jelinek. 1998. Exploiting syntactic structure for language modeling. In *Proc. COLING-ACL*, volume 1, Montreal, Canada.

N. Coccaro and D. Jurafsky. 1998. Towards better integration of semantic predictors in statistical language modeling. In *Proc. ICSLP-98*, volume 6, pages 2403–2406, Sydney.

L. Galescu and E. R. Ringger. 1999. Augmenting words with linguistic information for n-gram language models. In *Proc. 6th EuroSpeech*, Budapest, Hungary.

J. T. Goodman. 2001. A bit of progress in language modeling. *Microsoft Technical Report MSR-TR-2001-72.*

F. Jelinek. 1999. *Statistical methods for speech recognition.* The MIT Press.

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

C. Manning and H. Schutze. 1999. *Foundations of statistical natural language processing.* The MIT Press.

W. Wang and M. P. Harper. 2002. The superARV language model: Investigating the effectiveness of tightly integrating multiple knowledge sources. In *Proc. Conf. Empirical Methods in Natural Language Processing*, pages 238–247, Philadelphia.