COMBINATION OF LATENT SEMANTIC ANALYSIS BASED LANGUAGE MODELS FOR MEETING RECOGNITION

Michael Pucher
Telecommunications Research Center Vienna
Vienna, Austria
Speech and Signal Processsing Lab, TU Graz
Graz, Austria
email: pucher@ftw.at

Yan Huang and Özgür Çetin
International Computer Science Institute
Berkeley, USA
email: yan@icsi.berkeley.edu, ocetin@icsi.berkeley.edu

ABSTRACT
Latent Semantic Analysis (LSA) defines a semantic similarity space using a training corpus. This semantic similarity can be used for dealing with long distance dependencies, which are an inherent problem for traditional word-based n-gram models. Since LSA models adapt dynamically to topics, and meetings have clear topics, we conjecture that these models can improve speech recognition accuracy on meetings. This paper presents perplexity and word error rate results for LSA models for meetings. We present results for models trained on a variety of corpora including meeting data and background domain data, and for combinations of multiple LSA models together with a word-based n-gram model. We show that the meeting and background LSA models can improve over the baseline n-gram models in terms of perplexity and that some background LSA models can significantly improve over the n-gram models in terms of word error rate. For the combination of multiple LSA models we did however not see such an improvement.

KEY WORDS
Speech Recognition, Latent Semantic Indexing

1 Introduction

Word-based n-gram models are a popular and fairly sucessful paradigm in language modeling. With these models it is however difficult to model long distance dependencies which are present in natural language [1].

LSA maps a corpus of documents onto a semantic vector space. Long distance dependencies are modeled by representing the context or history of a word and the word itself as a vector in this space. The similarity between these two vectors is used to predict a word given a context. Since LSA models the context as a bag of words it has to be combined with n-gram models to include word-order statistics of the short span history. Language models that combine word-based n-gram models with LSA models have been successfully applied to conversational speech recognition and to the Wall Street Journal recognition task [2][3].

We conjecture that LSA based language models can also help to improve speech recognition for meetings, because meetings have clear topics and LSA models adapt dynamically to topics. Due to the sparseness of available data for language modeling for meetings it is important to combine meeting LSA models that are trained on relatively small corpora with background LSA models which are trained on larger corpora. The meeting domain is our adaptation domain and we have multiple background domains from broadcast news to web data.

2 LSA based Language Models

2.1 Constructing the Semantic Space

In LSA first the training corpus is encoded as a word–document co-ocurrence matrix W (using weighted term frequency). This matrix has high dimension and is highly sparse. Let $\mathcal{V}$ be the vocabulary with $|\mathcal{V}| = M$ and $\mathcal{T}$ be a text corpus containing n documents. Let c_{ij} be the number of occurrences of word i in document j, c_i the number of occurrences of word i in the whole corpus, i.e. $c_i = \sum_{j=1}^{N} c_{ij}$, and c_j the number of words in document j. The elements of W are given by

$$[W]_{ij} = (1 - \epsilon_{w_i}) \frac{c_{ij}}{c_j} \tag{1}$$

where ϵ_{w_i} is defined as

$$\epsilon_{w_i} = -\frac{1}{\log N} \sum_{j=1}^{N} \frac{c_{ij}}{c_i} \log \frac{c_{ij}}{c_i} \, . \tag{2}$$

ϵ_w will be used as a short-hand for ϵ_{w_i}. Informative words will have a low value of ϵ_w. Then a semantic space with much lower dimension is constructed using Singular Value Decomposition (SVD) [4].

$$W \approx \hat{W} = U \times S \times V^T \tag{3}$$

For some order $r \ll \min(m, n)$, U is a $m \times r$ left singular matrix, S is a $r \times r$ diagonal matrix that contains r singular values, and V is a $n \times r$ right singular matrix. The vector $u_i S$ represents word w_i, and $v_j S$ represents document d_j.

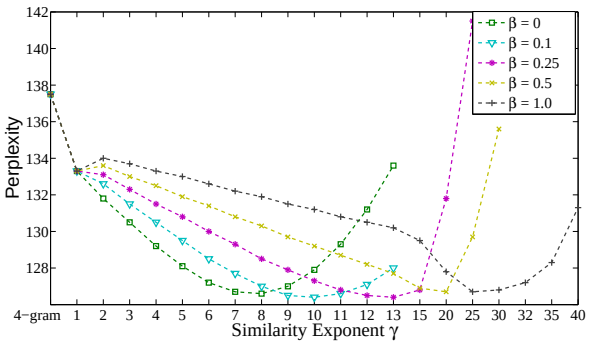

Figure 1. Perplexities for the Fisher LSA model with different γ and β values.

2.2 LSA Probability

In this semantic space the cosine similarity between words and documents is defined as

$$K_{\text{sim}}(w_i, d_j) \triangleq \frac{u_i S v_j^T}{||u_i S^{\frac{1}{2}}|| \cdot ||v_j S^{\frac{1}{2}}||} . \quad (4)$$

Since we need a probability for the integration with the n-gram models, the similarity is converted into a probability by normalizing it. According to [5], we extend the small dynamic range of the similarity function by introducing a temperature parameter γ. Our experiments show that the tuning of γ can lower the perplexity. Figure 1 shows perplexities for an LSA model trained on Fisher conversational speech data with different γ values compared to a 4-gram. Additionally an offset $\beta \in [0, 1]$ is added to the similarities to avoid pruning of similarities between $1 - \beta$ and 1. The exponent that mimimizes the perplexity increases with the offset, the minimum perplexity however stays the same.

We also have to define the concept of a pseudo-document d_{t-1} using the word vectors of all words preceding w_t, i.e. $w_1, \ldots, w_{t-1}$. This is needed because the model is used to compare words with documents that have not been seen so far. In the construction of the pseudo-document we also include a decay parameter $\delta < 1$ that renders words closer in the history more significant.

The conditional probability of a word w_t given a pseudo-document d_{t-1} is defined as

$$P_{\text{LSA}}(w_t|d_{t-1}) \triangleq \frac{[K_{\text{sim}}(w_t, d_{t-1}) - K_{\min}(d_{t-1})]^\gamma}{\sum_w [K_{\text{sim}}(w, d_{t-1}) - K_{\min}(d_{t-1})]^\gamma} \quad (5)$$

where $K_{\min}(d_{t-1}) = \min_w K(w, d_{t-1})$ to make the resulting similarities nonnegative [3].

2.3 Combining LSA and n-gram Models

For the interpolation of the word based n-gram models and the LSA models we used the methods defined in Table 1. λ is a fixed constant interpolation weight, and $\propto$ denotes that the result is normalized by the sum over the whole vocabulary. λ_w is a word-dependent parameter defined as

$$\lambda_w \triangleq \frac{1 - \epsilon_w}{2} . \quad (6)$$

This definition ensures that the n-gram model gets at least half of the weight. λ_w is higher for more informative words.

Model	Definition
n-gram (baseline)	$P_{n-\text{gram}}$
Linear interpolation (LIN)	$\lambda P_{\text{LSA}} + (1-\lambda) P_{n-\text{gram}}$
Similarity modulated n-gram interpolation (SIMMOD)	$\propto (K_{\text{sim}} - K_{\min}) P_{n-\text{gram}}$
Information weighted geometric mean interpolation (INFG)	$\propto P_{\text{LSA}}^{\lambda_w} P_{n-\text{gram}}^{1-\lambda_w}$

Table 1. *Interpolation methods.*

We used three different methods for the interpolation of n-gram models and LSA models. The *information weighted geometric mean*, the *similarity modulated n-gram* and simple *linear interpolation*. The *information weighted geometric mean* interpolation represents a loglinear interpolation of normalized LSA probabilities and the standard n-gram, weighted by λ_w. The *similarity modulated n-gram* interpolation uses K_{sim} and $K_{\min}$ directly, without normalizing first.

2.4 Combining LSA Models

For the combination of multiple LSA models we tried two different approaches. The first approach was the linear interpolation of LSA models with optimized λ_i where $\lambda_{n+1} = 1 - (\lambda_1 + \ldots + \lambda_n)$:

$$P_{\text{lin}} \triangleq \lambda_1 P_{\text{Lsa}_1} + \ldots + \lambda_n P_{\text{Lsa}_n} + \lambda_{n+1} P_{n-\text{gram}} \quad (7)$$

Our second approach was the INFG Interpolation with optimized θ_i where $\lambda_w^{(n+1)} = 1 - (\lambda_w^{(1)} + \ldots + \lambda_w^{(n)})$:

$$P_{\text{infg}} \propto P_{\text{Lsa}_1}^{\lambda_w^{(1)}\theta_1} \ldots P_{\text{Lsa}_n}^{\lambda_w^{(n)}\theta_n} P_{n-\text{gram}}^{\lambda_w^{(n+1)}\theta_{n+1}} \quad (8)$$

The parameter θ_i have to be optimized since the $\lambda_w^{(k)}$ depend on the corpus, so that a certain corpus can get a higher weight because of a content-word-like distribution of w, although the whole data does not well fit the meeting domain. In general we saw that the λ_w values were higher for the background domain models than for the meeting models. But taking the n-gram mixtures as an example the meeting models should get a higher weight than the background models. For this reason the λ_w of the background models have to be lowered using θ.

To ensure that the n-gram model gets a certain part α of the distribution, we define $\lambda_w^{(k)}$ for word w and LSA model $\mathrm{Lsa_k}$ as

$$\lambda_w^{(k)} \triangleq \frac{1-\epsilon_w^{(k)}}{\frac{n}{1-\alpha}} \quad (9)$$

where $\epsilon_w^{(k)}$ is the uninformativeness of word w in LSA model $\mathrm{Lsa_k}$ as defined in (2) and n is the number of LSA models. This is a generalization of definition (6). Through the generalization it is also possible to train α, the mimimum weight of the n-gram model.

For the INFG interpolation we had to optimize the model parameters θ_i, the part of the n-gram model α, and the γ exponent for each LSA model [6]. For the optimization meeting heldout data was used, containing four ICSI, four CMU, and four NIST meetings.

3 Meeting Models

3.1 Perplexities

For the training and testing of our first models we used the ICSI meeting corpus [7]. The training set contains 730K words. For this test we used the 2002 meeting evaluation development set (dev02) consisting of 37K words. We used the meeting boundaries as document boundaries, which are needed for the training of the LSA model.

Table 2 shows the perplexity results for ICSI meetings for the different methods. While the *linear interpolation* (LIN) and the *similarity modulated n-gram* (SIMMOD) do not bring any improvements over the baseline trigram model, the *information weighted geometric mean* (INFG) reduces perplexity. The improvement of the *information weighted geometric mean* interpolation over the trigram model is consistent with findings in [3]. For the other interpolations we always used the INFG method, since it outperformed all other interpolation methods.

Model	Perplexity
3-gram	84.3
INFG	81.7
SIMMOD	85.1
LIN	88.2

Table 2. *Perplexity results for ICSI meetings on dev02.*

The next meting model was trained on 880K words of CMU, ICSI, LDC and NIST meetings. For these and the other tests we used the 2004 NIST meeting development test set (dev04) consisting of 20K words. Table 3 shows the perplexities for this model interpolated using the INFG method with the n-gram model that was estimated with modified Kneser-Ney smoothing. CMU, ICSI, LDC and NIST meetings are the subsets of the data. There are small improvements for all meeting sites (CMU, ICSI, LDC, NIST).

Model	All	CMU	ICSI	LDC	NIST
4-gram	127.6	172.9	76.4	156.3	132.2
INFG	123.7	168.5	75.4	149.0	127.7

Table 3. *Perplexity results for all meetings on dev04.*

3.2 Word Error Rates

For our word error rate experiments we used test data from the NIST Spring 2005 Meeting Rich Transcription (RT-05S) evaluation [8], which contains meetings from several different sites.

	n-gram	LSA
AMI	25.5	25.5
CMU	24.7	24.9
ICSI	19.8	19.5
NIST	25.7	25.8
VT	27.0	26.9
ALL	24.9	24.8

Table 4. *Relative word error rate improvements for meeting LSA models.*

The relative WER improvements on the meeting data (Table 4) are neither significant for the ICSI data ($+1.5\%$, $p = 0.1$), nor for the VT data ($+0.3\%$, $p = 0.4$) or for the complete data set ($+0.4\%$, $p = 0.8$) according to a matched-pairs test.

4 Background Domain Models

4.1 Perplexities

Since the training corpora for meetings are very small we trained further LSA models on multiple background domains. A mixture of language models trained on adaptation and background domains has also been used for word-based n-gram models for meetings by [8]. The transcripts of the following widely-used corpora were used: Fisher, Hub4-LM96 and TDT4 (see Table 5). Furthermore we used data collected from the web, similar to CMU, ICSI and NIST meetings (=Meet-Web), and the Fisher corpus (=FWeb) as shown in Table 5. The document boundaries for the Fisher data were conversation sides, for the Hub4-LM96 broadcast news data they were news sites, and for the web data we used websites as document boundaries. The n-gram model used for meeting recognition in the 2005 NIST Meeting Speech Recognition Evaluations [8] was also trained on the above data. When we first interpolated our small meeting LSA models with this large n-gram mixture model we saw no improvements. This finding motivated us to include data from background domains.

Training Source	# of words ($\times 10^3$)
Fisher	23357
Hub4-LM96	130850
TDT4	11869
Meet-Web	147510
FWeb	530284

Table 5. *Training data sources.*

Model	Hub4-LM96	Tdt4	Meet-Web	Fisher
4-gram	144.1	238.8	145.5	131.5
INFG	134.2	224.6	137.4	123.9

Model	FWeba	FWebb	FWebc	FWebd
4-gram	130.0	130.0	130.0	130.0
INFG	123.3	124.1	123.5	123.8

Table 6. *Perplexity results on dev04.*

Table 6 shows the estimated perplexities for the models trained on the background domain data. The corresponding n-gram models where trained on the same data. We had to split the Fisher web data FWeb into four parts FWeba, FWebb, FWebc, and FWebd because it was too big to train one LSA model on it. The test set was again the dev04 test set. There are improvements over all background domains. We find these results promising for interpolating of multiple LSA models. Each LSA model is able to improve over the baseline n-gram model.

Concerning the γ exponent parameter defined in (5) that is used to expand the small dynamic range of the LSA similarity, we found that the optimal value of γ is higher for bigger models. The optimal value for the meeting model is 5, for the Fisher model it is 7 and for all other models it is 9, using an offset $\beta = 0$.

4.2 Word Error Rates

For the background domain models we got significant WER improvements for two of the data and meeting sites. For the Fisher data we got a relative WER improvement of $+1.1\%$ on the CMU data that is significant ($p = 0.05$). For the FWebc data we got a relative WER improvement of $+1.7\%$ on the ICSI data that is significant ($p = 0.04$).

5 Combined LSA Models

5.1 Perplexities

Perplexity results for the combination of all of the 8 background LSA models, the meeting LSA model, and the n-gram mixture model trained on all the available data are shown in Table 7.

Model	All	CMU	ICSI	LDC	NIST
4-gram	85.4	104.1	67.0	87.5	89.8
LIN	85.4	104.1	67.0	87.5	89.8
INFG	84.5	103.0	66.2	86.4	88.9

Table 7. *Perplexity results for combined LSA models on dev04.*

In case of the linear interpolation all LSA models get zero weight, so there is no improvement over the n-gram model. The INFG interpolation gives a small improvement, where the highest θ_i weights are given to the meeting LSA model, followed by the models trained on the Fisher and the Fweb data.

5.2 Word Error Rates

	n-gram	LIN	INFG
AMI	24.7	24.7	24.5
CMU	26.5	26.5	26.7
ICSI	22.6	22.6	22.7
NIST	24.4	24.4	24.4
VT	24.4	24.4	24.4
ALL	24.5	24.5	24.6

Table 8. *Relative word error rate improvements for combined LSA models.*

For the combination of LSA models using INFG interpolation (Table 8) we got a relative WER improvement on the AMI data of $+0.8\%$ which is not significant ($p = 0.2$).

6 Summary and Conclusion

When we first tried to interpolate the meeting LSA model with a mixture n-gram model, trained on all available data (Table 5) we did not see any improvements in perplexity. Our conclusion was that the LSA model does not capture more information than the n-gram model in this case. But when we were training on the same data we always saw an improvement of the LSA model over the n-gram model in terms of perplexity. This shows us that the LSA models captures some additional information compared to the n-gram model, if trained on the same data.

Since it is not feasible to train one LSA model on all the data our next step was to think about combinations of LSA models. The most promising technique was the log-linear INFG interpolation with optimized interpolation weights. But even with this interpolation we did only see small improvements in terms of perplexity. Concerning perplexity we can conclude that we can achieve improve-

ments over all background domains, but that we are still missing an interpolation method for multiple LSA models.

One possible problem in combining multiple LSA models could be that there are too many cases where the models give different similarites for the same words and contexts. In this case the word entropy could still make a difference with the INFG method. But with a similar entropy the models would neutralize each other.

Concerning the word error rates we also saw significant improvements for some of the background models, but no significant improvements for the combined LSA models and the meeting model. These results suggest a preselection of background models which are then used for the interpolation.

We showed that LSA based language models can decrease perplexity for meeting language modeling using a variety of background domain data, ranging from broadcast news and conversational speech to text collected from the web. We also discussed possible combinations of LSA models, which make it feasible to train these models on very large corpora.

Furthermore we showed that we can achieve significant improvements in terms of WER with some of the background domain models. For the combined LSA models we did however not see a significant improvement in terms of WER.

The significantly different results in terms of word error rate for different training corpora and meeting sites suggest that the LSA model combination should be preceded by an LSA based matching between meeting sites and training data.

7 Acknowledgements

This work was supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811) and by Kapsch Carrier-Com AG and Mobilkom Austria AG together with the Austrian competence centre programme **K***plus*.

References

[1] C. Chelba and F. Jelinek, Exploiting Syntactic Structure for Language Modeling, *Proceedings of COLING-ACL'98*, Montreal, Canada, 1998, 225-231.

[2] J. Bellegarda, Large Vocabulary Speech Recognition with Multispan Statistical Language Models, *IEEE Transactions on Speech and Audio Processing*, 8(1), 2000, 76-84.

[3] Y. Deng and S. Khudanpur, Latent Semantic Information in Maximum Entropy Language Models for Conversational Speech Recognition, *Proceedings of HLT-NAACL*, Edmonton, Canada, 2003, 56-63.

[4] S. Deerwester et.al., Indexing by Latent Semantic Analysis, *Journal of the American Society for Information Science*, 41(6), 1990, 391-407.

[5] N. Coccaro and D. Jurafsky, Towards better Integration of Semantic Predictors in Statistical Language Modeling, *Proceedings of ICSLP*, Sydney, 1998, 2403-2406.

[6] M. Pucher, Y. Huang, Ö. Çetin, Optimization of Latent Semantic Analysis based Language Model Interpolation for Meeting Recognition, *Proceedings of IS-LTC*, Ljubljana, Slovenia, 2006.

[7] A. Janin et.al., The ICSI Meeting Corpus, *Proc. Intl. Conf. on Acoustics, Speech and Signal Processing*, Hong Kong, 2003, 364-367.

[8] A. Stolcke et.al., Further Progress in Meeting Recognition: The ICSI-SRI Spring 2005 Speech-to-Text Evaluation System, *Proceedings of NIST MLMI Meeting Recognition Workshop*, Edinburgh, 2005, 463-475.