NOVEL TOPIC N-GRAM COUNT LM INCORPORATING DOCUMENT-BASED TOPIC DISTRIBUTIONS AND N-GRAM COUNTS

Md. Akmal Haidar and Douglas O'Shaughnessy

INRS-EMT, 6900-800 De La Gauchetiere Ouest, Montreal (Quebec), H5A 1K6, Canada

ABSTRACT

In this paper, we introduce a novel topic n-gram count language model (NTNCLM) using topic probabilities of training documents and document-based n-gram counts. The topic probabilities for the documents are computed by averaging the topic probabilities of words seen in the documents. The topic probabilities of documents are multiplied by the document-based n-gram counts. The products are then summed-up for all the training documents. The results are used as the counts of the respective topics to create the NTNCLMs. The NTNCLMs are adapted by using the topic probabilities of a development test set that are computed as above. We compare our approach with a recently proposed TNCLM [1], where the long-range information outside of the n-gram events is not encountered. Our approach yields significant perplexity and word error rate (WER) reductions over the other approach using the Wall Street Journal (WSJ) corpus.

Index Terms— Statistical n-gram language model, speech recognition, mixture models, topic models

1. INTRODUCTION

Statistical n-gram LMs have been used successfully for speech recognition and many other applications. They compute the probability of the n^{th} word by conditioning on the previous $n-1$ history (h) words. They suffer from the shortage of long-range information, which limits performance. To capture the long-range information, one of the earliest attempts was a cache-based LM that took advantage that a word observed earlier in a document could occur again. This helps to increase the probability of the seen words when predicting the next word [2]. A similar idea was used in trigger-based LM adaptation, which uses a maximum entropy approach [3] to raise the probability of unseen but topically related words. In addition recently, latent topic analysis has been used broadly to compensate for the weaknesses of n-gram models. Several techniques such as Latent Semantic Analysis (LSA) [4,5], Probabilistic Latent Semantic Analysis (PLSA) [6,7], and latent Dirichlet allocation (LDA) [8] have been studied to extract the latent semantic information from a training corpus. The LDA model has been used successfully in recent research work for LM adaptation [1, 9–15]. In [10], a unigram scaling approach is used for the LDA adapted unigram model to minimize the distance between the adapted model and the background model [10]. In [1], a topic n-gram count LM was proposed where the topic probabilities of n-grams were created by using features of the LDA model.

In this paper, we extend our previous work [1] to incorporate the long-range useful information outside of n-gram events. In [1], TNCLMs were formed by computing the topic probabilities of background n-grams $P(t_k|w_1,\dots,w_n), (k = 1,\dots,K)$ by averaging the topic probabilities of the words $P(t_k|w_i)$ present in the n-grams. $P(t_k|w_1,\dots,w_n)$ were multiplied with the global count of the n-gram $C(h,w_i)$ and then used as the counts of the topics to create the TNCLMs. However, the TNCLMs do not capture the long-range important information outside of the n-gram events. Here, we propose a novel TNCLM (NTNCLM) where $P(t_k|w_1,\dots,w_n)$ are derived by using the topic probabilities of the training documents $P(t_k|d_l), (l = 1,\dots,M)$. $P(t_k|d_l)$ are calculated by averaging the $P(t_k|w_i)$ for words seen in the documents. $P(t_k|d_l)$ are multiplied with the document-based n-gram counts $C(h,w_i,d_l)$ and then summed-up for all training documents. The results are used as the counts of topics to create the NTNCLMs. The TNCLMs and NTNCLMs are both adapted by using the topic mixture weights obtained by averaging the $P(t_k|w_i)$ over the seen words of a development test set d_t. The adapted models are interpolated with a background tri-gram model to capture the local lexical regularities. The complete idea is described in Figure 1. In the figure, $\gamma_{k,n}$ and γ_{k,d_l} represent $P(t_k|w_1,\dots,w_n)$ and $P(t_k|d_l)$ respectively. N_{d_l} and N_{d_t} describe the the number of words seen in the training document d_l and the development test set d_t. We compare our approach with an adapted n-gram LM obtained by unsupervised language model adaptation using latent semantic marginals [10] and the interpolation of the adapted TNCLM with the background model [1]. We apply the LM adaptation approaches after the first pass decoding and have seen that our approach outperforms the conventional approaches.

The rest of this paper is organized as follows. Section 2 is used for reviewing the LDA model. TNCLM generation is described in section 3. The proposed NTNCLM generation is explained in section 4. Section 5 is used to illustrate the LM adaptation approaches. The experimental setup and re-

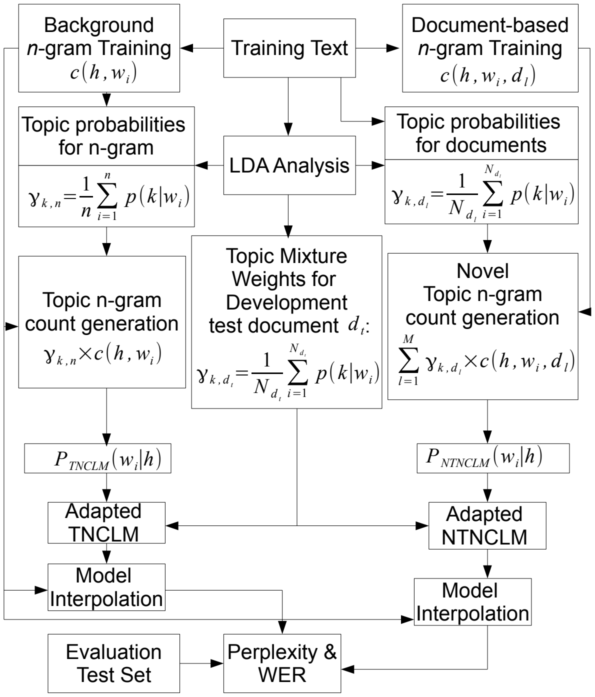
Background n-gram Training $c(h,w_i)$
Training Text
Document-based n-gram Training $c(h,w_i,d_l)$
Topic probabilities for n-gram
$\gamma_{k,n}=\frac{1}{n}\sum_{i=1}^{n} p(k|w_i)$
LDA Analysis
Topic probabilities for documents
$\gamma_{k,d_l}=\frac{1}{N_{d_l}}\sum_{i=1}^{N_{d_l}} p(k|w_i)$
Topic n-gram count generation
$\gamma_{k,n}\times c(h,w_i)$
Topic Mixture Weights for Development test document d_l:
$\gamma_{k,d_l}=\frac{1}{N_{d_l}}\sum_{i=1}^{N_{d_l}} p(k|w_i)$
Novel Topic n-gram count generation
$\sum_{l=1}^{M}\gamma_{k,d_l}\times c(h,w_i,d_l)$
$P_{TNCLM}(w_i|h)$
$P_{NTNCLM}(w_i|h)$
Adapted TNCLM
Adapted NTNCLM
Model Interpolation
Model Interpolation
Evaluation Test Set
Perplexity & WER

Fig. 1. Adaptation of TNCLM and NTNCLM

sults are described in section 6. Finally the conclusions are described in section 7.

2. LATENT DIRICHLET ALLOCATION

LDA is a generative probabilistic topic model for documents in a corpus. Documents are represented by the random latent topics[1], which are characterized by a distribution over words. The model can be described as follows:

- Each document $d_l = w_1, \ldots, w_{N_{d_l}}$ is generated as a mixture of unigram models, where the topic mixture weight θ_{d_l} is drawn from a prior Dirichlet distribution with parameter α:

$$p(\theta_{d_l}|\alpha) = \frac{\Gamma(\sum_{k=1}^{K}\alpha_k)}{\prod_{k=1}^{K}\Gamma(\alpha_k)}\theta_{d_{l1}}^{\alpha_1-1}\ldots\theta_{d_{lK}}^{\alpha_K-1} \qquad (1)$$

- For each word in document d_l:
  - Choose a topic t_k from the multinomial distribution θ_{d_l}.
  - Choose a word w_i from the multinomial distribution $P(w_i|t_k,\beta)$,

where $\alpha = \{\alpha_1, \ldots, \alpha_K\}$ is used as the representation count for the K latent topics, θ_{d_l} indicates the relative importance of topics for the document d_l and $p(w_i|t_k,\beta)$ represents the word probabilities conditioned on the topic with a Dirichlet prior β and indicates the relative importance of particular words in a topic t_k.

The probability of the document can be estimated by marginalizing unobserved variables θ_{d_l} and t_k as:

$$P(d_l|\alpha,\beta) = \int p(\theta_{d_l}|\alpha)\prod_{i=1}^{N_{d_l}}\sum_{k=1}^{K} p(t_k|d_l,\theta_{d_l})P(w_i|t_k,\beta)d\theta_{d_l} \qquad (2)$$

2.1. LDA Training

The parameters of the LDA model are computed by using the MATLAB topic modeling toolbox [16,17]. Here, we obtain a word-topic matrix WP and a document topic matrix DP. An entry $WP(w_i,t_k)$ describes the number of times the word w_i has been assigned to topic t_k over the training set. An entry $DP(d_l,t_k)$ of the DP matrix contains the total occurrences of words in document d_l that are from a topic t_k. We used the above matrices to compute the probability of words given topics and the probability of topics given documents as [17, 18]:

$$P(w_i|t_k,\beta) = \frac{WP(w_i,t_k)+\beta}{WP(.,t_k)+V\beta}, \qquad (3)$$

$$P(t_k|d_l,\theta_{d_l}) = \frac{DP(d_l,t_k)+\alpha}{DP(d_l,.)+K\alpha}, \qquad (4)$$

where $WP(w_i,t_k)$ is the number of occurrences of word w_i in topic t_k, $WP(.,t_k)$ is the total count of words in topic t_k, $DP(d_l,t_k)$ holds the total occurrences of words from topic t_k in document d_l, $DP(d_l,.)$ contains the occurrences of words from all topics in document d_l and V is the total number of words.

3. TOPIC N-GRAM COUNT LANGUAGE MODEL

The features of the LDA model were used to create the topic n-gram count language model (TNCLM) [1]. Because of bag-of-words characteristics of the LDA model, each word has equal weight in determining the topic mixtures. Also, latent topics are independent of each other in the LDA topic set. A constraint was taken such that the total count of an n-gram for all topics is equal to the count of that n-gram in the training set. The probability of topic t_k given word w_i, $P(t_k|w_i)$ and the probability of word w_i given topic t_k, $P(w_i|t_k)$ were used as confidence measures in determining the topic probability of the n-grams, where $P(t_k|w_i)$ outperforms $P(w_i|t_k)$ [1]. In this paper, we used only the confidence measure $P(t_k|w_i)$.

The topic probabilities of the background n-grams are computed using the average topic probability of words in the n-grams. These probabilities are normalized and then multiplied by the global counts of the n-grams and finally used as

[1] Topics are unobserved in LDA

the counts of the n-grams for the corresponding topics. The topic probabilities for each n-gram are computed as [1]:

$$P(t_k|w_1,\ldots,w_n) = \frac{1}{n}\sum_{i=1}^{n} P(t_k|w_i), \tag{5}$$

where $P(t_k|w_1,\ldots,w_n)$ is the probability of the n-gram in topic t_k. $P(t_k|w_i)$ is computed using the Bayes's formula:

$$P(t_k|w_i) = \frac{P(w_i|t_k)P(t_k)}{\sum_{k=1}^{K} P(w_i|t_k)P(t_k)}, \tag{6}$$

where $P(t_k)$ is the prior topic probability and $P(w_i|t_k)$ (Equation 3) is the word probability in topic t_k. $P(t_k)$ is computed as:

$$P(t_k) = \frac{\sum_i WP(w_i,t_k) + \beta V}{\sum_k(\sum_i WP(w_i,t_k) + \beta V)}. \tag{7}$$

The topic probabilities for each n-gram are then normalized so that the total topic probabilities for each n-gram are summed to one and then the topic probabilities are multiplied with the original count of that n-gram in the training set. For example, a tri-gram “A B C” is seen 20 times in the training corpus and for 4 topics, the topic probabilities of the tri-gram “A B C” are 0.2, 0.3, 0.1 and 0.4, which are computed using equation 5. Therefore, the counts for the “A B C” in 4 topics are 4, 6, 2 and 8. However, the results of the multiplication are the topic n-gram counts for the corresponding topics. The topic n-gram language models are then generated using the topic n-gram counts and defined as TNCLMs [1].

4. PROPOSED NTNCLM

The TNCLM model does not capture the information outside the n-gram events as it directly uses topic probabilities of words $P(t_k|w_i)$ in generating topic probability of n-grams $P(t_k|w_1,\ldots,w_n)$. To compensate for the weakness of this model, we introduce a novel TNCLM (NTNCLM) that uses topic probabilities of training documents $P(t_k|d_l)$ in computing topic probabilities of n-grams $P(t_k|w_1,\ldots,w_n)$.

The topic probabilities of the training documents d_l ($l = 1,\ldots,M$) are created by averaging the topic probability of words present in the respective documents as:

$$P(t_k|d_l) = \frac{1}{N_{d_l}}\sum_{i=1}^{N_{d_l}} P(t_k|w_i), \tag{8}$$

where N_{d_l} is the number of words seen in training document d_l. The topic probabilities for each document d_l are then normalized so that the total topic probabilities for each document are summed to one.

The topic probability for an n-gram is created as:

$$\begin{aligned} P(t_k|w_1,\ldots,w_n) &= \sum_{l=1}^{M} P(t_k|d_l)P(d_l|w_1,\ldots,w_n) \\ &= \sum_{l=1}^{M} P(t_k|d_l)\frac{C(w_1,\ldots,w_n,d_l)}{C(w_1,\ldots,w_n)}, \end{aligned} \tag{9}$$

The topic probability of the n-gram is then multiplied with the global n-gram count $C(w_1,\ldots,w_n)$ and the product is used as the count of the n-gram for the respective topic. The results can be written as:

$$\begin{aligned} C(w_1,\ldots,w_n,t_k) &= P(t_k|w_1,\ldots,w_n) * C(w_1,\ldots,w_n) \\ &= \sum_{l=1}^{M} P(t_k|d_l)C(w_1,\ldots,w_l,d_l), \end{aligned} \tag{10}$$

where $P(t_k|d_l)$ are the topic probabilities for training documents created by Equation 8. The NTNCLMs are then created by using the respective topic n-gram counts. We also introduce other TNCLMs defined as LDA TNCLMs (LTNCLMs) by using Equation 10 where the $P(t_k|d_l)$ is computed by using the document-topic matrix DP (Equation 4).

5. LM ADAPTATION APPROACH

In the LDA model, a document can be generated by a mixture of topics. So, for a test document $d_t = w_1,\ldots,w_{N_{d_t}}$, the dynamically adapted topic model by using a mixture of LMs from different topics is computed as:

$$P_{ATNCLM/ANTNCLM/ALTNCLM}(w_i|h) = \sum_{k=1}^{K} \delta_k P_i(w_i|h), \tag{11}$$

where $P_k(w_i|h)$ is the k^{th} TNCLM/NTNCLM/LTNCLM, $P_{ATNCLM/ANTNCLM/ALTNCLM}(w_i|h)$ are the adapted n-gram count LMs and δ_k is the k^{th} topic mixture weight. The mixture weights for the TNCLMs and NTNCLMs are computed as:

$$P(k|d_t) = \frac{1}{N_{d_t}}\sum_{i=1}^{N_{d_t}} P(k|w_i), \tag{12}$$

where N_{d_t} is the number of words seen in the development test document d_t. For the LTNCLMs, the mixture weights are computed using LDA inference [18].

The ATNCLM/ANTNCLM/ALTNCLMs are then interpolated with the background (B) n-gram model to capture the local constraints using linear interpolation as:

$$\begin{aligned} P_L(w_i|h) &= \lambda P_B(w_i|h) + \\ &\quad (1-\lambda)P_{ATNCLM/ANTNCLM/ALTNCLM}(w_i|h), \end{aligned} \tag{13}$$

where λ is an interpolation weight.

6. EXPERIMENTS

6.1. Data and experimental setup

We used the Wall Street Journal (WSJ) corpus [19] to evaluate the LM adaptation approaches. The SRILM toolkit [20] and the HTK toolkit [21] are used for generating the LMs and computing the WER respectively. The '87-89 WSJ corpus is used to train the tri-gram background (B) model and the tri-gram TNCLMs/NTNCLMs/LTNCLMs using the back-off version of the Witten-Bell smoothing. To reduce the computational cost, we incorporated the cutoffs 1 and 3 on the background bi-gram and background tri-gram counts respectively. The Witten-Bell smoothing from the SRILM toolkit is used as the TNCLMs/NTNCLMs/LTNCLMs are generated using the floating counts. The LDA and the closed vocabulary language models are trained using the 5K non-verbalized punctuation closed vocabulary. We define the α and β for LDA analysis as 50/K and 0.01 respectively [17, 18]. The acoustic model from [22] is used in our experiments. The acoustic model is trained by using all WSJ and TIMIT [23] training data, the 40 phones set of the CMU dictionary [24], approximately 10000 tied-states, 32 gaussians per state and 64 gaussians per silence state. The acoustic waveforms are parameterized into a 39-dimensional feature vector consisting of 12 cepstral coefficients plus the 0^{th} cepstral, delta and delta delta coefficients, normalized using cepstral mean subtraction ($MFCC_{0-D-A-Z}$). We evaluated the cross-word models. The values of the word insertion penalty, beam width, and the language model scale factor are -4.0, 350.0, and 15.0 respectively [22]. The development and the evaluation test sets are the **si_dt_05.odd** (248 sentences from 10 speakers) and the Nov'93 Hub 2 5K test data from the ARPA November 1993 WSJ evaluation (215 sentences from 10 speakers) [19, 25]. The topic mixture weights δ and the interpolation weight λ are tuned on the development test set. The results are noted on the evaluation test set.

6.2. Experimental Results

We tested our proposed approaches for various topic sizes. The perplexity results of the models are explained in Table 1.

Table 1. Perplexity results of the language models

Language Model	25 Topics	50 Topics
Background (B)	83.4	83.4
ATNCLM	105.5	134.0
ALTNCLM	86.5	111.4
ANTNCLM	86.2	110.4
B+ATNCLM	75.3	75.6
B+ALTNCLM	74.6	74.8
B+ANTNCLM	74.7	74.9

From Table 1, we can note that the proposed approaches outperform the ATNCLM [1] in both stand-alone and interpolated form for all topic sizes.

We also evaluated the LM adaptation approaches for speech recognition. We used the unigram scaling approach of the LDA adapted model (LDA unigram scaling) [10] and the interpolation of background models with the ATNCLM model [1] for comparison. We evaluated the WER experiments using lattice rescoring. In the first pass, we used the background tri-gram language model for lattice generation. In the second pass, interpolation of the background and the adapted models are applied for lattice rescoring. The experimental results are plotted in Figure 2. From Figure 2, we can note that the proposed B+ANTNCLM gives significant WER reductions of about 9.9% (8.1% to 7.3%), 7.6% (7.9% to 7.3%), 3.9% (7.6% to 7.3%), and 1.4% (7.4% to 7.3%) for 25 topics, and about 7.4% (8.1% to 7.5%), 5.1% (7.9% to 7.5%), 3.8% (7.8% to 7.5%), and 1.3% (7.6% to 7.5%) for 50 topics over the background trigram, LDA unigram scaling [10], B+ATNCLM [1] and B+ALTNCLM (also proposed by us) approaches respectively. However, the B+ALTNCLM model outperforms the background, LDA unigram scaling [10] and B+ATNCLM [1] approaches respectively. The significance improvement in WER using the proposed B+ANTNCLM is done by using a match-pair-test where the misrecognized words in each test utterance are counted. We obtain the P-values of 0.03 and 0.02 relative to B+ATNCLM [1] for the topic sizes 25 and 50 respectively. At a significance level of 0.05, our proposed B+ANTNCLM model outperforms the B+ATNCLM model [1].

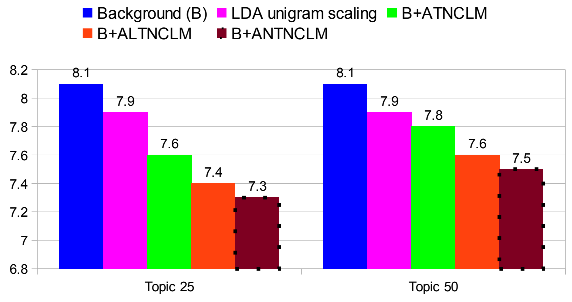

Fig. 2. WER results of the language models

7. CONCLUSIONS

In this paper, we proposed a novel TNCLM (NTNCLM) using document-based topic distributions and n-gram counts. The topic probabilities for training documents are created by averaging the confidence measure (topic probability given words) of the words present in the documents. Then, they are multiplied by the document-based n-gram counts and the products are summed up for all the training documents. The results are used as the n-gram counts for the respective topics to create the NTNCLM. We also introduce an LDA TNCLM (LTNCLM) as above where the topic probabilities for documents are created by using the document-topic matrix obtained from the LDA model training. We compare our approaches with a recently proposed TNCLM [1], which uses the above confidence measures to compute the probability of background n-grams and is used as the count of the n-grams for the respective topics. The normalized topic probabilities of the n-gram are multiplied by the global n-gram count to form the topic n-gram count for the respective topics. However, TNCLM does not capture the long-range information outside of the n-gram events. To compensate for the weaknesses of the TNCLMs, the NTNCLMs and LTNCLMs are proposed here. Both TNCLMs, NTNCLMs and LTNCLMs are adapted and then interpolated with a background trigram model to capture the short-range information. The proposed approaches yield better performance over the conventional approaches.

REFERENCES

[1] M. A. Haidar and D. O'Shaughnessy, "Topic N-gram count language model for speech recognition," in *Proceedings of IEEE Spoken Language Technology (SLT) Workshop*, 2012, pp. 165–169.

[2] R. Kuhn and R. D. Mori, "A cache-based natural language model for speech recognition," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, vol. 12 (6), pp. 570–583, 1990.

[3] R. Rosenfeld, "A maximum entropy approach to adaptive statistical language modeling," *Computer Speech and Language*, vol. 10 (3), pp. 187–228, 1996.

[4] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41(6), pp. 391 – 407, 1990.

[5] J. R. Bellegarda, "Exploiting latent semantic information in statistical language modeling," *IEEE Transactions on Speech and Audio Processing*, vol. 88 (8), pp. 1279–1296, 2000.

[6] T. Hofmann, "Probabilistic latent semantic analysis," in *Proceedings of the Fifteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, San Francisco, CA, 1999, pp. 289–296, Morgan Kaufmann.

[7] D. Gildea and T. Hofmann, "Topic-based language models using EM," in *Proceedings of EUROSPEECH*, 1999, pp. 2167–2170.

[8] D. M. Blei, A. Y. Ng., and M. I. Jordan, "Latent Dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[9] Y.-C. Tam and T. Schultz, "Dynamic language model adaptation using variational bayes inference," in *Proceedings of INTERSPEECH*, 2005, pp. 5–8.

[10] Y.-C. Tam and T. Schultz, "Unsupervised language model adaptation using latent semantic marginals," in *Proceedings of INTERSPEECH*, 2006, pp. 2206–2209.

[11] F. Liu and Y. Liu, "Unsupervised language model adaptation incorporating named entity information," in *Proceedings of ACL*, 2007, pp. 672–679.

[12] F. Liu and Y. Liu, "Unsupervised language model adaptation via topic modeling based on named entity hypothesis," in *Proceedings of ICASSP*, 2008, pp. 4921–4924.

[13] M. A. Haidar and D. O'Shaughnessy, "Novel weighting scheme for unsupervised language model adaptation using latent Dirichlet allocation," in *Proceedings of INTERSPEECH*, 2010, pp. 2438–2441.

[14] M. A. Haidar and D. O'Shaughnessy, "Unsupervised language model adaptation using N-gram weighting," in *Proceedings of CCECE*, 2011, pp. 857–860.

[15] M. A. Haidar and D. O'Shaughnessy, "LDA-based LM adaptation using latent semantic marginals and minimum discrimination information," in *Proceedings of EUSIPCO*, 2012, pp. 2040–2044.

[16] M. Steyvers, "MATLAB topic modeling toolbox," `http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm`, 2013.

[17] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *National Academy of Science*, vol. 101 (1), pp. 5228–5235, 2004.

[18] G. Heinrich, *Parameter estimation for text analysis*, Technical report, Fraunhofer IGD, 2009.

[19] -, *CSR-II (WSJ1) Complete*, Linguistic Data Consortium, Philadelphia, 1994.

[20] A. Stolcke, "SRILM-an extensible language modeling toolkit," in *Proceedings of ICSLP*, 2002, pp. 901–904.

[21] S. Young, P. Woodland, G. Evermann, and M. Gales, "The HTK toolkit 3.4.1.," `http://htk.eng.cam.ac.uk/`, 2013.

[22] K. Vertanen, "HTK wall street journal training recipe," `http://www.keithv.com/software/htk/us/`, 2013.

[23] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "TIMIT acoustic-phonetic continuous speech corpus," *Linguistic Data Consortium*, 1993.

[24] -, "The Carnegie Mellon University (CMU) pronounciation dictionary," `http://www.speech.cs.cmu.edu/cgi-bin/cmudict`, 2013.

[25] P.C. Woodland, J. J. Odell, V. Valtchev, and S. J. Young, "Large vocabulary continuous speech recognition using HTK," in *Proceedings of ICASSP*, 1994, pp. 125–128.