Adaptive Language Modeling for Word Prediction

Keith Trnka
University of Delaware
Newark, DE 19716
trnka@cis.udel.edu

Abstract

We present the development and tuning of a topic-adapted language model for word prediction, which improves keystroke savings over a comparable baseline. We outline our plans to develop and integrate style adaptations, building on our experience in topic modeling to dynamically tune the model to both topically and stylistically relevant texts.

1 Introduction

People who use Augmentative and Alternative Communication (AAC) devices communicate slowly, often below 10 words per minute (wpm) compared to 150 wpm or higher for speech (Newell et al., 1998). AAC devices are highly specialized keyboards with speech synthesis, typically providing single-button input for common words or phrases, but requiring a user to type letter-by-letter for other words, called fringe vocabulary. Many commercial systems (e.g., PRC's ECO) and researchers (Li and Hirst, 2005; Trnka et al., 2006; Wandmacher and Antoine, 2007; Matiasek and Baroni, 2003) have leveraged word prediction to help speed AAC communication rate. While the user is typing an utterance letter-by-letter, the system continuously provides potential completions of the current word to the user, which the user may select. The list of predicted words is generated using a language model.

At best, modern devices utilize a trigram model and very basic recency promotion. However, one of the lamented weaknesses of ngram models is their sensitivity to the training data. They require substantial training data to be accurate, and increasingly more data as more of the context is utilized. For example, Lesher et al. (1999) demonstrate that bigram and trigram models for word prediction are not saturated even when trained on 3 million words, in contrast to a unigram model. In addition to the problem of needing substantial amounts of training text to build a reasonable model, ngrams are sensitive to the difference between training and testing/user texts. An ngram model trained on text of a different topic and/or style may perform very poorly compared to a model trained and tested on similar text. Trnka and McCoy (2007) and Wandmacher and Antoine (2006) have demonstrated the domain sensitivity of ngram models for word prediction.

The problem of utilizing ngram models for conversational AAC usage is that no substantial corpora of AAC text are available (much less conversational AAC text). The most similar available corpora are spoken language, but are typically much smaller than written corpora. The problem of corpora for AAC is that similarity and availability are inversely related, illustrated in Figure 1. At one extreme, a very large amount of formal written English is available, however, it is very dissimilar from conversational AAC text, making it less useful for word prediction. At the other extreme, logged text from the current conversation of the AAC user is the most highly related text, but it is extremely sparse. While this trend is demonstrated with a variety of language modeling applications, the problem is more severe for AAC due to the extremely limited availability of AAC text. Even if we train our models on both a large number of general texts in addition to highly related in-domain texts to address the problem, we

Proceedings of the ACL-08: HLT Student Research Workshop (Companion Volume), pages 61–66,
Columbus, June 2008.

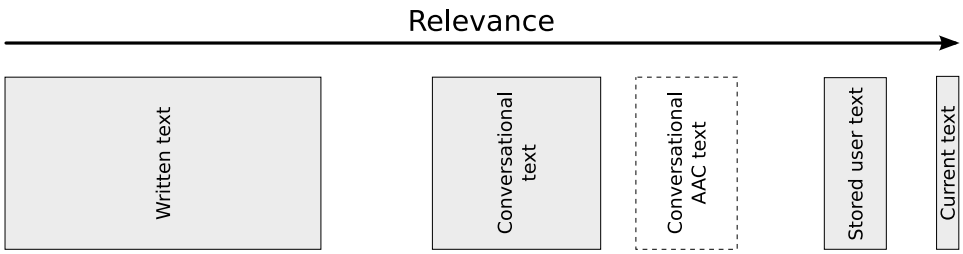

Figure 1: The most relevant text available is often the smallest, while the largest corpora are often the least relevant for AAC word prediction. This problem is exaggerated for AAC.

must focus the models on the most relevant texts.

We address the problem of balancing training size and similarity by dynamically adapting the language model to the most topically relevant portions of the training data. We present the results of experimenting with different topic segmentations and relevance scores in order to tune existing methods to topic modeling. Our approach is designed to seamlessly degrade to the baseline model when no relevant topics are found, by interpolating frequencies as well as ensuring that all training documents contribute some non-zero probabilities to the model. We also outline our plans to adapt ngram models to the style of discourse and then combine the topical and stylistic adaptations.

1.1 Evaluating Word Prediction

Word prediction is evaluated in terms of keystroke savings — the percentage of keystrokes saved by taking full advantage of the predictions compared to letter-by-letter entry.

$$KS = \frac{keys_{\textit{letter-by-letter}} - keys_{\textit{with prediction}}}{keys_{\textit{letter-by-letter}}} \times 100\%$$

Keystroke savings is typically measured automatically by simulating a user typing the testing data of a corpus, where any prediction is selected with a single keystroke and a space is automatically entered after selecting a prediction. The results are dependent on the quality of the language model as well as the number of words in the prediction window. We focus on 5-word prediction windows. Many commercial devices provide optimized input for the most common words (called core vocabulary) and offer word prediction for all other words (fringe vocabulary). Therefore, we limit our evaluation to fringe words only, based on a core vocabulary list from conversations of young adults.

We focus our training and testing on Switchboard, which we feel is similar to conversational AAC text. Our overall evaluation varies the training data from Switchboard training to training on out-of-domain data to estimate the effects of topic modeling in real-world usage.

2 Topic Modeling

Topic models are language models that dynamically adapt to testing data, focusing on the most related topics in the training data. It can be viewed as a two stage process: 1) identifying the relevant topics by scoring and 2) tuning the language model based on relevant topics. Various other implementations of topic adaptation have been successful in word prediction (Li and Hirst, 2005; Wandmacher and Antoine, 2007) and speech recognition (Bellegarda, 2000; Mahajan et al., 1999; Seymore and Rosenfeld, 1997). The main difference of the topic modeling approach compared to Latent Semantic Analysis (LSA) models (Bellegarda, 2000) and trigger pair models (Lau et al., 1993; Matiasek and Baroni, 2003) is that topic models perform the majority of generalization about topic relatedness at testing time rather than training time, which potentially allows user text to be added to the training data seamlessly.

Topic modeling follows the framework below

$$P_{\textit{topic}}(w \mid h) = \sum_{t \in \textit{topics}} P(t \mid h) * P(w \mid h, t)$$

where w is the word being predicted/estimated, h represents all of the document seen so far, and t represents a single topic. The linear combination for topic modeling shows the three main areas of variation in topic modeling. The posterior probability,

$P(w \mid h, t)$ represents the sort of model we have; how topic will affect the adapted language model in the end. The prior, $P(t \mid h)$, represents the way topic is identified. Finally, the meaning of $t \in topics$, requires explanation — what is a topic?

2.1 Posterior Probability — Topic Application

The topic modeling approach complicates the estimation of probabilities from a corpus because the additional conditioning information in the posterior probability $P(w \mid h, t)$ worsens the data sparseness problem. This section will present our experience in lessening the data sparseness problem in the posterior, using examples on trigram models.

The **posterior probability** requires more data than a typical ngram model, potentially causing **data sparseness** problems. We have explored the possibility of estimating it by geometrically combining a topic-adapted unigram model (i.e., $P(w \mid t)$) with a context-adapted trigram model (i.e., $P(w \mid w_{-1}, w_{-2})$), compared to straightforward measurement ($P(w \mid w_{-1}, w_{-2}, t)$). Although the first approach avoids the additional data sparseness, it makes an assumption that the topic of discourse only affects the vocabulary usage. Bellegarda (2000) used this approach for LSA-adapted modeling, however, we found this approach to be inferior to direct estimation of the posterior probability for word prediction (Trnka et al., 2006). Part of the reason for the lesser benefit is that the overall model is only affected slightly by topic adaptations due to the tuned exponential weight of 0.05 on the topic-adapted unigram model. We extended previous research by forcing trigram predictions to occur over bigrams and so on (rather than backoff) and using the topic-adapted model for re-ranking within each set of predictions, but found that the forced ordering of the ngram components was overly detrimental to keystroke savings.

Backoff models for topic modeling can be constructed either before or after the linear interpolation. If the backoff is performed after interpolation, we must also choose whether smoothing (a prerequisite for backoff) is performed before or after the interpolation. If we smooth before the interpolation, then the frequencies will be overly discounted, because the smoothing method is operating on a small fraction of the training data, which will reduce the benefit of higher-order ngrams in the overall model. Also, if we combine probability distributions from each topic, the combination approach may have difficulties with topics of varying size. We address these issues by instead combining frequencies and performing smoothing and backoff after the combination, similar to Adda et al. (1999), although they used corpus-sized topics. The advantage of this approach is that the held-out probability for each distribution is appropriate for the training data, because the smoothing takes place knowing the number of words that occurred in the whole corpus, rather than for each small segment. This is especially important when dealing with small and different sized topics.

The **linear interpolation affects smoothing methods negatively** — because the weights are less than one, the combination decreases the total sum of each conditional distribution. This will cause smoothing methods to underestimate the reliability of the models, because smoothing methods estimate the reliability of a distribution based on the absolute number of occurrences. To correct this, after interpolating the frequencies we found it useful to scale the distribution back to its original sum. The scaling approach improved keystroke savings by 0.2%–0.4% for window size 2–10 and decreased savings by 0.1% for window size 1. Because most AAC systems provide 5–7 predictions, we use this approach. Also, because some smoothing methods operate on frequencies, but the combination model produces real-valued weights for each word, we found it necessary to bucket the combined frequencies to convert them to integers.

Finally, we required an **efficient smoothing** method that could discount each conditional distribution individually to facilitate on-demand smoothing for each conditional distribution, in contrast to a method like Katz' backoff (Katz, 1987) which smoothes an entire ngram model at once. Also, Good-Turing smoothing proved too cumbersome, as we were unable to rely on the ratio between words in given bins and also unable to reliably apply regression. Instead, we used an approximation of Good-Turing smoothing that performed similarly, but allowed for substantial optimization.

2.2 Prior Probability — Topic Identification

The topic modeling approach uses the current testing document to tune the language model to the most relevant training data. The benefit of adaptation is dependent on the quality of the similarity scores. We will first present our representation of the current document, which is compared to unigram models of each topic using a similarity function. We determine the weight of each word in the current document using frequency, recency, and topical salience.

The **recency of use** of a word contributes to the relevance of the word. If a word was used somewhat recently, we would expect to see the word again. We follow Bellegarda (2000) in using an exponentially decayed cache with weight of 0.95 to model this effect of recency on importance at the current position in the document. The weight of 0.95 represents a preservation in topic, but with a decay for very stale words, whereas a weight of 1 turns the exponential model into a pure frequency model and lower weights represent quick shifts in topic.

The importance of each word occurrence in the current document is a factor of not just its frequency and recency, but also it's **topical salience** — how well the word discriminates between topics. For this reason, we decided to use a technique like Inverse Document Frequency (IDF) to boost the weight of words that occur in only a few documents and depress the weights of words that occur in most documents. However, instead of using IDF to measure topical salience, we use Inverse Topic Frequency (ITF), which is more specifically tailored to topic modeling and the particular kinds of topics used.

We evaluated several **similarity functions** for topic modeling, initially using the cosine measure for similarity scoring and scaling the scores to be a probability distribution, following Florian and Yarowsky (1999). The intuition behind the cosine measure is that the similarity between two distributions of words should be independent of the length of either document. However, researchers have demonstrated that cosine is not the best relevance metric for other applications, so we evaluated two other topical similarity scores: Jacquard's coefficient, which performed better than most other similarity measures in a different task for Lee (1999) and Naïve Bayes, which gave better results than cosine in topic-adapted language models for Seymore and Rosenfeld (1997). We evaluated all three similarity metrics using Switchboard topics as the training data and each of our corpora for testing using cross-validation. We found that cosine is consistently better than both Jacquard's coefficient and Naïve Bayes, across all corpora tested. The differences between cosine and the other methods are statistically significant at $p < 0.001$. It may be possible that the ITF or recency weighting in the cache had a negative interaction with Naïve Bayes; traditionally raw frequencies are used.

We found it useful to **polarize the similarity scores**, following Florian and Yarowsky (1999), who found that transformations on cosine similarity reduced perplexity. We scaled the scores such that the maximum score was one and the minimum score was zero, which improved keystroke savings somewhat. This helps fine-tune topic modeling by further boosting the weights of the most relevant topics and depressing the weights of the less relevant topics.

Smoothing the scores helps prevent some scores from being zero due to lack of word overlap. One of the motivations behind using a linear interpolation of all topics is that the resulting ngram model will have the same coverage of ngrams as a model that isn't adapted by topic. However, the similarity score will be zero when no words overlap between the topic and history. Therefore we decided to experiment with similarity score smoothing, which records the minimum nonzero score and then adds a fraction of that score to all scores, then only apply upscaling, where the maximum is scaled to 1, but the minimum is not scaled to zero. In pilot experiments, we found that smoothing the scores did not affect topic modeling with traditional topic clusters, but gave minor improvements when documents were used as topics.

Stemming is another alternative to improving the similarity scoring. This helps to reduce problems with data sparseness by treating different forms of the same word as topically equivalent. We found that stemming the cache representations was very useful when documents were treated as topics (0.2% increase across window sizes), but detrimental when larger topics were used (0.1–0.2% decrease across window sizes). Therefore, we only use stemming when documents are treated as topics.

2.3 What's in a Topic — Topic Granularity

We adapt a language model to the most relevant *topics* in training text. But what is a topic? Traditionally, document clusters are used for topics, where some researchers use hand-crafted clusters (Trnka et al., 2006; Lesher and Rinkus, 2001) and others use automatic clustering (Florian and Yarowsky, 1999). However, other researchers such as Mahajan et al. (1999) have used each individual document as a topic. On the other end of the spectrum, we can use whole corpora as topics when training on multiple corpora. We call this spectrum of topic definitions *topic granularity*, where manual and automatic document clusters are called *medium-grained* topic modeling. When topics are individual documents, we call the approach *fine-grained* topic modeling. In fine-grained modeling, topics are very specific, such as seasonal clothing in the workplace, compared to a medium topic for clothing. When topics are whole corpora, we call the approach *coarse-grained* topic modeling. Coarse-grained topics model much more high-level topics, such as research or news.

The results of testing on Switchboard across different topic granularities are showin in Table 1. The in-domain test is trained on Switchboard only. Out-of-domain training is performed using all other corpora in our collection (a mix of spoken and written language). Mixed-domain training combines the two data sets. Medium-grained topics are only presented for in-domain training, as human-annotated topics were only available for Switchboard. Stemming was used for fine-grained topics, but similarity score smoothing was not used due to lack of time.

The topic granularity experiment confirms our earlier findings that topic modeling can significantly improve keystroke savings. However, the variation of granularity shows that the size of the topics has a strong effect on keystroke savings. Human annotated topics give the best results, though fine-grained topic modeling gives similar results without the need for annotation, making it applicable to training on not just Switchboard but other corpora as well. The coarse grained topic approach seems to be limited to finding acceptable interpolation weights between very similar and very dissimilar data, but is poor at selecting the most relevant corpora from a collection of very different corpora in the out-of-domain test. Another problem may be that many of the corpora are only homogeneous in style but not topic. We would like to extend our work in topic granularity to testing on other corpora in the future.

3 Future Work – Style and Combination

Topic modeling balances the similarity of the training data against the size by tuning a large training set to the most topically relevant portions. However, keystroke savings is not only affected by the topical similarity of the training data, but also the stylistic similarity. Therefore, we plan to also adapt models to the style of text. Our success in adapting to the topic of conversation leads us to believe that a similar process may be applicable to style modeling — splitting the model into style identification and style application. Because we are primarily interested in syntactic style, we will focus on part of speech as the mechanism for realizing grammatical style. As a pilot experiment, we compared a collection of our technical writings on word prediction with a collection of our research emails on word prediction, finding that we could observe traditional trends in the POS ngram distributions (e.g., more pronouns and phrasal verbs in emails). Therefore, we expect that distributional similarity of POS tags will be useful for style identification. We envision a single style s affecting the likelihood of each part of speech p in a POS ngram model like the one below:

$$P(w \mid w_{-1},w_{-2},s) = \sum_{p \in POS(w)} P(p \mid p_{-1},p_{-2},s) * P(w \mid p)$$

In this reformulation of a POS ngram model, the prior is conditioned on the style and the previous couple tags. We will use the overall framework to combine style identification and modeling:

$$P_{style}(w \mid h) = \sum_{s \in styles} P(s \mid h) * P(w \mid w_{-1},w_{-2},s)$$

The topical and stylistic adaptations can be combined by adding topic modeling into the style model shown above. The POS posterior probability $P(w \mid p)$ can be additionally conditioned on the topic of discourse. Topic identification and the topic summation would be implemented consistently with the standalone topic model. Also, the POS framework

Model type	In-domain	Out-of-domain	Mixed-domain
Trigram baseline	60.35%	53.88%	59.80%
Switchboard topics (medium grained)	61.48% (+1.12%)	–	–
Document as topic (fine grained)	61.42% (+1.07%)	54.90% (+1.02%)	61.17% (+1.37%)
Corpus as topic (coarse grained)	–	52.63% (-1.25%)	60.62% (+0.82%)

Table 1: Keystroke savings across different granularity topics and training domains, tested on Switchboard. Improvement over baseline is shown in parentheses. All differences from baseline are significant at $p < 0.001$

facilitates cache modeling in the posterior, allowing direct adaptation to the current text, but with less sparseness than other context-aware models.

4 Conclusions

We have created a topic adapted language model that utilizes the full training data, but with focused tuning on the most relevant portions. The inclusion of all the training data as well as the usage of frequencies addresses the problem of sparse data in an adaptive model. We have demonstrated that topic modeling can significantly increase keystroke savings for traditional testing as well as testing on text from other domains. We have also addressed the problem of annotated topics through fine-grained modeling and found that it is also a significant improvement over a baseline ngram model. We plan to extend this work to build models that adapt to both topic and style.

Acknowledgments

This work was supported by US Department of Education grant H113G040051. I would like to thank my advisor, Kathy McCoy, for her help as well as the many excellent and thorough reviewers.

References

Gilles Adda, Michèle Jardino, and Jean-Luc Gauvain. 1999. Language modeling for broadcast news transcription. In *Eurospeech*, pages 1759–1762.

Jerome R. Bellegarda. 2000. Large vocabulary speech recognition with multispan language models. *IEEE Transactions on Speech and Audio Processing*, 8(1):76–84.

Radu Florian and David Yarowsky. 1999. Dynamic Nonlocal Language Modeling via Hierarchical Topic-Based Adaptation. In *ACL*, pages 167–174.

Slava M. Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics Speech and Signal Processing*, 35(3):400–401.

R. Lau, R. Rosenfeld, and S. Roukos. 1993. Trigger-based language models: a maximum entropy approach. In *ICASSP*, volume 2, pages 45–48.

Lillian Lee. 1999. Measures of distributional similarity. In *ACL*, pages 25–32.

Gregory Lesher and Gerard Rinkus. 2001. Domain-specific word prediction for augmentative communication. In *RESNA*, pages 61–63.

Gregory W. Lesher, Bryan J. Moulton, and D. Jeffery Higgonbotham. 1999. Effects of ngram order and training text size on word prediction. In *RESNA*, pages 52–54.

Jianhua Li and Graeme Hirst. 2005. Semantic knowledge in word completion. In *ASSETS*, pages 121–128.

Milind Mahajan, Doug Beeferman, and X. D. Huang. 1999. Improved topic-dependent language modeling using information retrieval techniques. In *ICASSP*, volume 1, pages 541–544.

Johannes Matiasek and Marco Baroni. 2003. Exploiting long distance collocational relations in predictive typing. In *EACL-03 Workshop on Language Modeling for Text Entry*, pages 1–8.

Alan Newell, Stefan Langer, and Marianne Hickey. 1998. The rôle of natural language processing in alternative and augmentative communication. *Natural Language Engineering*, 4(1):1–16.

Kristie Seymore and Ronald Rosenfeld. 1997. Using Story Topics for Language Model Adaptation. In *Eurospeech*, pages 1987–1990.

Keith Trnka and Kathleen F. McCoy. 2007. Corpus Studies in Word Prediction. In *ASSETS*, pages 195–202.

Keith Trnka, Debra Yarrington, Kathleen McCoy, and Christopher Pennington. 2006. Topic Modeling in Fringe Word Prediction for AAC. In *IUI*, pages 276–278.

Tonio Wandmacher and Jean-Yves Antoine. 2006. Training Language Models without Appropriate Language Resources: Experiments with an AAC System for Disabled People. In *LREC*.

T. Wandmacher and J.Y. Antoine. 2007. Methods to integrate a language model with semantic information for a word prediction component. In *EMNLP*, pages 506–513.