

INTEGRATING A CONTEXT-DEPENDENT PHRASE GRAMMAR IN THE VARIABLE N-GRAM FRAMEWORK

Manhung Siu, Mari Ostendorf †

EEE Department, Hong Kong University of Science and Technology, Clearwater Bay, Hong Kong
eemsiu@ust.hk

† Dept. of Electrical Engineering, University of Washington, Seattle, WA USA
mo@ee.washington.edu

ABSTRACT

This paper focuses on the learning of multi-word lexical units, or phrases, and how to model them within the variable n-gram framework. We introduce the notion of context-dependent phrases and suggest an algorithm for unsupervised learning of phrases. Also, we propose an approach to integrate a phrase grammar and a variable n-gram without the need of explicitly handling multi-word lexical items. The combined variable n-gram phrase grammar improves recognition accuracy on the Switchboard corpus over both the baseline trigram and using a variable n-gram alone.

1. INTRODUCTION

Although words in English are reasonable lexical units for language modeling, there are many cases that longer lexical units may be more appropriate. Frequently used word sequences, such as *I mean* or *you know*, are so common in conversational speech that they may be effectively used by the speaker as a single lexical item. We call these multi-word units “phrases”.

There are several ways of treating a multi-word sequence: 1) as context-independent phrases for which the same sequence is always treated as a phrase in all contexts, such as *a lot of*; 2) as context-dependent phrases for which a word sequence is treated as a phrase only in some particular contexts; and 3) either as word sequences or as phrase units in the same context non-deterministically, context dependently or independently. An example of a context-dependent phrase is in the sentence *That’s really about you know his work*, where we can specify that in all occasions where *you know* is preceded by *about*, it is treated as a phrase. In this paper, we address the problem of phrase selection assuming deterministic but context-dependent phrases. Once the phrases are selected, the extension to the non-deterministic case is straightforward [1].

Learning and modeling multi-word lexical units (phrases) has been the subject of considerable interest in recent years. Brown *et al.* [2] proposed an algorithm for learning phrases by computing the ratio of the word bigram probability $p(w_y|w_x)$ to the unigram probability $p(w_y)$ and

selecting those pairs which have a high ratio as phrases. More recent studies that use phrases in language modeling are done on structured domains such as the air traffic control (ATC) [3] or the air travel information system (ATIS) domains [4], [5]. In some cases, these phrases are selected by hand and are represented as non-terminals of hand-written grammar rules [3]. The work reported in [6] creates phrases based on a set of context-free grammar (CFG) rules. Then, words or phrases are replaced by CFG non-terminals and are clustered together. In other cases, they are automatically learned, such as selecting phrases that minimize the perplexity in cross-validation [7]. Non-deterministic phrases are reported in [8], but using only phrase unigrams. Although progress has been made using phrases in language modeling, almost all the reported work is done on constrained domains. Furthermore, the notion of context-dependent phrases has not been looked at.

In this paper, we will focus on the learning of context-dependent phrases and implementation of a phrase grammar within the variable n-gram framework. In Section 2, we describe the effect of adding a two-word phrase to a bigram model. In Section 3, we describe the connections between a phrase grammar and a variable n-gram. Section 4 covers phrase selection. Experimental results are reported in Section 5, and we conclude in Section 6 with a brief summary of the key points.

2. EFFECT OF A TWO-WORD PHRASE

Let us consider $W = w_1, w_2, w_3, w_4, w_5$, a five-word sequence and denote $p(\cdot)$ and $\hat{p}(\cdot)$ as the maximum likelihood bigram probability estimates before and after a context-dependent two-word phrase $w_{3,4}$ is formed in the context of w_2 by concatenating the words w_3 and w_4 . $p(W)$ and $\hat{p}(W)$ can be expressed as,

$$\begin{aligned} p(W) &= p(w_5|w_4)p(w_4|w_3)p(w_3|w_2)p(w_2|w_1)p(w_1) \\ \hat{p}(W) &= \hat{p}(w_5|w_{3,4})\hat{p}(w_{3,4}|w_2)p(w_2|w_1)p(w_1). \end{aligned}$$

Comparing the above equations, we notice that two new probability estimates, $\hat{p}(w_5|w_{3,4})$ and $\hat{p}(w_{3,4}|w_2)$ are needed. Furthermore, any bigram estimates conditioning

on w_2 , w_3 and w_4 , including $p(w_x|w_3)$ and $p(w_x|w_4)$ will be affected. However, $p(w_3|w_x)$ and $p(w_4|w_x)$ will not be affected because the phrase w_3, w_4 is defined only in the context of w_2 . To simplify the presentation, we use the maximum likelihood bigram estimation. Use of back-off will be discussed later; extensions to trigrams are straightforward.

Because the phrase $w_{3,4}$ is defined only in the context of w_2 , its count $c(w_{3,4}) = c(w_2, w_3, w_4)$. Thus, $\hat{p}(w_x|w_{3,4})$ can simply be computed by the ratio of counts given by

$$\hat{p}(w_x|w_{3,4}) = \frac{c(w_{3,4}, w_x)}{c(w_{3,4})} = \frac{c(w_2, w_3, w_4, w_x)}{c(w_2, w_3, w_4)}. \quad (1)$$

The bigram probability $\hat{p}(w_x|w_2)$, is the same as $p(w_x|w_2)$ except when $w_x = w_{3,4}$ and $w_x = w_3$. That is,

$$\hat{p}(w_x|w_2) = \begin{cases} \frac{c(w_2, w_3) - c(w_2, w_3, w_4)}{c(w_2)} & \text{if } w_x = w_3 \\ \frac{c(w_2, w_3, w_4)}{c(w_2)} & \text{if } w_x = w_{3,4} \\ p(w_x|w_2) & \text{otherwise.} \end{cases} \quad (2)$$

When we create the phrase $w_{3,4}$, we partition the counts of w_3 and w_4 into those that are part of the phrase and those that are not. The new bigram distribution conditioned on w_3 can be written as

$$\hat{p}(w_x|w_3) = \begin{cases} \frac{c(w_3, w_x)}{c(w_3) - c(w_2, w_3, w_4)} & \text{if } w_x \neq w_4 \\ \frac{c(w_3, w_4) - c(w_2, w_3, w_4)}{c(w_3) - c(w_2, w_3, w_4)} & \text{if } w_x = w_4. \end{cases} \quad (3)$$

Let us define the ratio of the new estimates to the old estimate, denoted $\alpha(w_3)$, where

$$\begin{aligned} \alpha(w_3) &= \frac{\hat{p}(w_x|w_3)}{p(w_x|w_3)} > 1 \\ &= \frac{c(w_3)}{c(w_3) - c(w_2, w_3, w_4)}. \end{aligned} \quad (4)$$

By using $\alpha(w_3)$, $p(w_x|w_3)$ can be written as

$$\hat{p}(w_x|w_3) = \begin{cases} \alpha(w_3)p(w_x|w_3) & \text{if } w_x \neq w_4 \\ \frac{c(w_3, w_4) - c(w_2, w_3, w_4)}{c(w_3) - c(w_2, w_3, w_4)} & \text{if } w_x = w_4. \end{cases} \quad (5)$$

Similarly, the new bigram estimate for words conditioned on w_4 is given by

$$\hat{p}(w_x|w_4) = \frac{c(w_4, w_x) - c(w_2, w_3, w_4, w_x)}{c(w_4) - c(w_2, w_3, w_4)}. \quad (6)$$

3. COMBINING PHRASE GRAMMAR AND VARIABLE N-GRAM

The term $\hat{p}(w_5|w_{3,4})$ is reminiscent of a variable n-gram [9, 10]. Therefore, it is interesting to consider the relationship between a variable n-gram and a phrase grammar. Although both extend the context of the sequence, the ways

they achieve this are very different. For the variable n-gram, the focus is on modeling the distributions given the history, $p(w_x|h)$, by extending or reducing the history h for the whole distribution. For the phrase model, the focus is on extending the context for particular words. In a sense, the phrase grammar is the dual of the variable n-gram. In the variable n-gram, extending the context affects only one distribution, but all observations. In the phrase grammar, creating a phrase affects all conditional distributions of the word but only a subset of the observations. In this section, we describe how we implement phrase grammar within the variable n-gram framework.

3.1. Implementing phrase grammar as a special variable n-gram

Since the phrase grammars and variable n-gram models both have the effect of extending the context of some distributions, it is advantageous for us to implement the phrase grammar as a special form of variable n-gram. By representing a phrase grammar as a special variable n-gram, we obtained better back-offs from phrase n-gram to word n-gram. To achieve this, we need to express all the phrase probabilities $\hat{p}(w_{3,4}|w_2)$ in terms of simple single word conditional distributions denoted $\tilde{p}(\cdot|w_3, w_2)$ and $\tilde{p}(\cdot|w_2)$. $\tilde{p}(\cdot|w_3, w_2)$ and $\tilde{p}(\cdot|w_2)$ must satisfy the following constraints:

$$\begin{aligned} \tilde{p}(w_x|w_3, w_2)\tilde{p}(w_3|w_2) &= \\ &\begin{cases} \frac{c(w_2, w_3, w_x) - c(w_2, w_3, w_4)}{c(w_2, w_3) - c(w_2, w_3, w_4)} & \text{if } w_x = w_4, \\ \hat{p}(w_x|w_3)\hat{p}(w_3|w_2) & \text{if } w_x \neq w_4. \end{cases} \end{aligned} \quad (7)$$

Define $\beta(w_2, w_3; w_4)$ as the ratio of the new counts of w_2, w_3 compared to their original counts. Then

$$\beta(w_2, w_3; w_4) = \frac{c(w_2, w_3) - c(w_2, w_3, w_4)}{c(w_2, w_3)}, \quad (8)$$

and

$$1 - \beta(w_2, w_3; w_4) = \frac{c(w_2, w_3, w_4)}{c(w_2, w_3)}. \quad (9)$$

By expanding $\hat{p}(w_x|w_3)$ and $\hat{p}(w_3|w_2)$ and rewriting the Equation 7, we obtained

$$\begin{aligned} \tilde{p}(w_x|w_3, w_2)\tilde{p}(w_3|w_2) &= \\ &\begin{cases} (1 - \beta(w_2, w_3; w_4))p(w_3|w_2) & \text{if } w_x = w_4, \\ \frac{c(w_3, w_x)}{c(w_3) - c(w_2, w_3, w_4)} \times & \\ \beta(w_2, w_3; w_4)p(w_3|w_2) & \text{if } w_x \neq w_4. \end{cases} \end{aligned} \quad (10)$$

This means that

$$\tilde{p}(\cdot|w_2) = p(\cdot|w_2)$$

and

$$\begin{aligned} \tilde{p}(w_x|w_3, w_2) &= \\ &\begin{cases} (1 - \beta(w_2, w_3; w_4)) & w_x = w_4, \\ \beta(w_2, w_3; w_4)\alpha(w_2, w_3)p(w_x|w_3) & w_x \neq w_4. \end{cases} \end{aligned} \quad (11)$$

4. SELECTING PHRASES

One criteria for selecting phrases is the improvement in likelihood. However, computing the exact likelihood changes for all possible phrase candidates is expensive. Instead, within the variable n-gram framework, we can assume that any effect of phrase grammar that affects only the context length are captured by the variable n-gram. So, the change of likelihood can for the phrase w_3, w_4 in the context of w_2 , denoted as $\delta_l(w_3, w_4; w_2)$, can be written as,

$$\begin{aligned} \delta_l(w_3, w_4; w_2) = & \\ & c(w_2, w_3, w_4) \log\left(\frac{\tilde{p}(w_4|w_3; w_2)}{p(w_4|w_3)}\right) \\ & + c(w_2, w_3) \beta(w_2, w_3; w_4) \log \beta(w_2, w_3; w_4) \\ & + c(w_3) \alpha(w_3) \log \alpha(w_3) \\ & + c(w_3, w_4) \gamma(w_3, w_4; w_2) \log(\gamma(w_3, w_4; w_2)), \end{aligned} \quad (12)$$

where $\gamma(w_3, w_4; w_2)$ is defined as

$$\gamma(w_3, w_4; w_2) = \frac{c(w_3, w_4) - c(w_2, w_3, w_4)}{c(w_3, w_4)}.$$

4.1. Multiple Phrases

Suppose we are creating two context-dependent phrases, w_2, w_3, w_x and w_2, w_3, w_y , then we need to compute the joint β and α denoted by $\hat{\beta}$ and $\hat{\alpha}$. $\beta(w_2, w_3)$, which is the ratio of the new counts of w_2, w_3 to its original count can be re-written as

$$\hat{\beta}(w_2, w_3; w_x, w_y) = \frac{c(w_2, w_3) - c(w_2, w_3, w_x) - c(w_2, w_3, w_y)}{c(w_2, w_3)}.$$

$\alpha(w_3)$, which is the ratio of the original counts of w_3 to the new counts, can also be re-written as

$$\hat{\alpha}(w_3) = \frac{c(w_3)}{c(w_3) - c(w_2, w_3, w_x) - c(w_2, w_3, w_y)} \quad (13)$$

The first three terms of Equation 12 can be computed easily with the updated α and β . Unfortunately, they do not decompose into a sum of contributions from each individual phrase. This implies that selecting the optimal phrases can be computationally expensive because all combinations have to be tried. Instead, a greedy algorithm can be used to select the most promising phrase first.

4.2. Simplifications

We can make further simplification by assuming that the effect on the unigram is small, i.e. $\alpha(w_3) = 1$. Furthermore, if we also assume that portion of w_3, w_4 in context is small comparing to w_3, w_4 not in context, i.e. $\gamma(w_3, w_4; w_2) = 1$, then, we obtain the following simplified algorithm.

1. Look at $p(w_x|w_2, w_3)$ and compute the weighted difference between $\tilde{p}(w_x|w_2, w_3)$ and $p(w_x|w_3)$ for all w_x , i.e.

$$score(w_x) = p(w_x|w_2, w_3) \log \frac{p(w_x|w_2, w_3)}{p(w_x|w_3)},$$

2. Define s_i rank ordered version of $score(w_x)$, such that $s_i > s_{i+1}$ and w_{x_i} the associated words.
3. Compute $\hat{\beta}_i(w_2, w_3)$ for all the words jointly up-to word w_{x_i}

4. Compute final score

$$f_i = s_i + p(w_x|w_2, w_3) \hat{\beta}_i(w_2, w_3) \log \hat{\beta}_i(w_2, w_3)$$

5. Compare $\max_i f_i$ with a threshold and select the $w_{x_1} \dots w_{x_i}$ to be expanded as phrases.

4.3. Back-off

In our derivation above, we do not include the back-off to simplify the equation. In the experiments, we use the Witten-Bell back-off, because its simple form is easy to incorporate in the formulas above. For the newly created distributions that are used to represent the phrase grammar, we have taken steps to ensure that the final probability after back-off is equal to the true phrase grammar probability.

4.4. Leave-one-out likelihood

Since building a phrase grammar is implicitly extending the context and increasing the number of parameters in the model, training likelihood will increase irrespective of how bad the phrase is. To compensate for this as well as to use the same framework as with variable n-gram design [9], we use the leave-one-out likelihood to evaluate $\delta_l(w_3, w_4)$ [11].

5. EXPERIMENTS AND RESULTS

Large vocabulary speech recognition experiments are conducted using the BBN Byblos system. The acoustic models used by the Byblos system are trained with 2 million words and 140 hours of conversational speech from the switchboard corpus. The data is segmented into sentences based on speaker turns, long pauses and non-speech events such as laughs and breath noise. The dictionary includes phonetic transcriptions of 26000 unique words, including some non-speech events (e.g. noise) modeled by a few special non-speech phones. The conversations are collected from more than 200 speakers. An acoustic model is trained for each gender. Each phone is modeled by 5 HMM states. Each state in the context-independent phone PTM system

Experiment	Perplexity	WER
Baseline 3-gram	118	39.57
Variable 4-gram	108	39.40
Variable n-gram + phrase	107	39.15

Table 1: Re-scoring results using the variable n-gram in combination with the phrase grammar.

is modeled by a Gaussian mixture model with 256 components, each modeled by a 45 dimensional mean and a diagonal covariance matrix. Vocal tract length normalization is used but not speaker-adapted training.

Our test set consists of seven conversations containing 6500 words from 35 minutes of speech. They are a subset of the 1995 NIST-administered large vocabulary conversational speech recognition evaluation set. The test speech is first decoded using Byblos recognizer with a trigram language model. An N-best list of 100 is generated for each sentence. Associated with each N-best hypothesis is five different “scores” provided by the recognizer. They are, the acoustic score, the language modeling score, the number of words, the number of silences, the number of phones. We perform our re-scoring by replacing the original language modeling score by the new language modeling score generated by either a variable n-gram or phrase grammar.

The phrase grammar is built by selecting any phrase that improves the delta leave-one-out log likelihood. This phrase grammar is built in combination with a baseline variable n-gram (to be consistent with the variable n-gram recognition). In Table 1, we show the re-scoring results and the perplexity number using the phrase models in combination with the variable n-gram. By comparing row 3 and row 2, we notice that incorporation of the phrase grammar improves the variable n-gram by 0.25% absolute.

6. SUMMARY

In this paper, we have introduced the notion of context-dependent phrase grammar. By analyzing the effect of introducing phrases, we introduced an algorithm that can learn what phrase to model automatically. We also showed that the phrase grammar can be combined with a variable n-gram by representing the phrase probability as a special form of a variable n-gram, which allows for a better back-off from phrase bigram to word bigram. This combination gives a small improvement in recognition performance compared to using the variable n-gram alone with only a 5% increase in the number of parameters. The places where a phrase grammar can help a variable n-gram are when the variable n-gram finds two distributions similar, but one or two particular words are not. For example, in the context of *those*, the variable n-gram decides that the distri-

bution $p(\cdot|those\ kind)$ and $p(\cdot|kind)$ are similar, but the phrase grammar picks out that *kind of* should be modeled as a phrase. Thus, phrase selection may provide a means of learning different senses of multiword sequences.

7. REFERENCES

- [1] M. Siu, *Learning local lexical structure in spontaneous speech language modeling*, Boston University Ph.D. thesis, 1998.
- [2] P. Brown, V. Pietra, P. V. deSouza, J. Lai, and R. L. Mercer, “Class-based n-gram models of natural language,” *Computational Linguistic*, 18:467–479, 1992.
- [3] M. Meteer and R. Rohlicek, “Statistical language modeling combining n-gram and context-free grammars,” In *Proc. Inter. Conf. on Acoust., Speech and Signal Proc.*, volume 2, pp. 173–176, 1993.
- [4] S. Seneff, H. Meng, and V. Zue, “Language modeling for recognition and understanding using layered bigrams,” In *Proc. Inter. Conf. on Spoken Language Processing*, pp. 317–320, 1992.
- [5] R. Moore et al, “Combining linguistic and structural knowledge sources,” In *Proc. ARPA Spoken Language Technology Workshop*, pp. 261–264, 1995.
- [6] M. K. McCandless and J. R. Glass, “Empirical acquisition of language models for speech recognition,” In *Proc. Inter. Conf. on Spoken Language Processing*, pp. 835–838, 1994.
- [7] K. Ries, F. D. Buo, and A. Waibel, “Class phrase models for language modeling,” In *Proc. Inter. Conf. on Spoken Language Processing*, pp. 398–401, 1996.
- [8] S. Deligne and F. Bimbot, “Language modeling by variable-length sequence: theoretical formulation and evaluation of multigrams,” In *Proc. Inter. Conf. on Spoken Language Processing*, pp. 169–172, 1995.
- [9] M. Siu and M. Ostendorf, “Variable n-grams and extensions for conversational speech language modeling,” *IEEE Trans. on Speech and Audio Processing*, to appear Jan 2000.
- [10] T. Niesler and P. Woodland, “Variable-length category n-gram language models,” *Computer Speech and Language*, vol. 21, pp. 1-26, 1999.
- [11] R. Kneser and H. Ney, “Improved clustering techniques for class-based statistical language modeling,” In *Proc. European Conference on Speech Comm. and Tech.*, pp. 973–976, 1993.