ANALYSING WEAKNESSES OF LANGUAGE MODELS FOR SPEECH RECOGNITION

Joerg P. Ueberla

Forum Technology - DRA Malvern
St. Andrews Road, Malvern,
Worcestershire, WR14 3PS, UK
email: ueberla@signal.dra.hmg.gb

ABSTRACT

In this paper, we propose to analyse the weaknesses of language models for speech recognition, in order to subsequently improve the models. First, a definition of a weakness of a language model that is applicable to almost all currently used models is given. This definition is then applied to a class based bi-gram model. The results show that one can gain considerable insight into a model by analysing its weaknesses. Moreover, when the model was modified in order to avoid one of the weaknesses, the modeling of unknown words, the performance of the model improved significantly.

1. INTRODUCTION

Since it is usually easier to improve a model once its weaknesses [1] are known, we propose to analyse the weaknesses of probabilistic language models used in speech recognition. To that end, a definition of a "weakness of a language model" is developed in section 2, and then applied to a concrete language model in section 3. Conclusions from this work follow in section 4.

2. WEAKNESSES OF LANGUAGE MODELS

In speech recognition, one is given a sequence of acoustic observations A and one tries to find the word sequence W^* that is most likely to correspond to A. In order to minimise the average probability of error, one should, according to Bayes' decision rule ([1, p.17]), choose

$$W^* = argmax_W p(W|A). \quad (1)$$

Based on Bayes' formula (see for example [2, p.150]), one can rewrite the probability from the right hand side of equation 1 according to the following equation:

$$p(W|A) = \frac{p(W) * p(A|W)}{p(A)}. \quad (2)$$

$p(W)$ is the probability that the word sequence W is spoken, $p(A|W)$ is the conditional probability that the acoustic signal A is observed when W is spoken and $p(A)$ is the probability of observing the acoustic signal A. Based on this formula, one can rewrite the maximization of equation 1 as

$$W^* = argmax_W \frac{p(W) * p(A|W)}{p(A)}. \quad (3)$$

Since $p(A)$ is the same for all W, the factor $p(A)$ does not influence the choice of W and maximising equation 3 is equivalent to maximising

$$W^* = argmax_W p(W) * p(A|W). \quad (4)$$

The component of the speech recogniser that calculates $p(A|W)$ is called the acoustic model, the component calculating $p(W)$ the language model.

Given that the task of the language model is to derive $p(W)$, how can one define a weakness of a language model? The first point to note is that the definition of a weakness should be related to the performance measure used to evaluate a language model. If they are not related, one can still identify and remove a weakness, but by doing so, one may not improve the performance of the model because the weakness is not related to the performance measure. Before defining a weakness, we therefore turn to the standard measure used to evaluate the performance of a language model.

The standard yardstick for comparing language models is the perplexity (see [3]), which is just the reciprocal of the geometric mean ("the average") of the probabilities a language model assigns to a sequence of words in a testing text. If $W = w_1, ..., w_i, ..., w_n$ denotes the

[1] The term "weakness" seems to be more adequate than the term "error", since we are dealing with probability distributions.

0-7803-2431-5/95 $4.00 © 1995 IEEE

words in a text, the total probability TP and the perplexity PP are

$$TP = p(W) = \prod_{i=1}^{i=n} p(w_i|w_1, ..., w_{i-1}) \quad (5)$$

$$PP = (TP)^{-\frac{1}{n}}. \quad (6)$$

For a large sample of text, the total probability TP can get extremely small. Therefore, from a practical point of view, it is more convenient to use the logarithm of the total probability LTP and the logarithm of the perplexity LP [2]:

$$LTP = log_2(TP) = \sum_{i=1}^{i=n} log_2(p(w_i|w_1, ..., w_{i-1}))(7)$$

$$LP = log_2(PP) = -\frac{1}{n} log_2(TP) = -\frac{1}{n} LTP. \ (8)$$

One can now describe a weakness of a language model in terms of the logarithm of the total probability LTP. Intuitively, a weakness of a language model is any part of the model that causes a large fraction of the LTP. In the following, this intuitive description will be formalised.

The testing text W can also be denoted by its index set $I_W = \{1, ..., n\}$. This way, one can denote any subset W_1 of words of W by giving the subset of indices $I_{W_1} \subseteq I_W$. For a given subset W_1, one can easily determine the LTP it causes (LTP_{W_1}) by summing up the logarithm of the probabilities of all the words in W_1:

$$LTP_{W_1} = \sum_{i \in W_1} log_2(p(w_i|w_1, ..., w_{i-1})). \quad (9)$$

Given LTP_{W_1}, one can then calculate the fraction of LTP caused by W_1 (f_{W_1}) as

$$f_{W_1} = \frac{LTP_{W_1}}{LTP}. \quad (10)$$

It is clear that one needs to improve the language model's prediction of the words that cause a large fraction of LTP, if one wants to improve the overall performance significantly.

Given the fraction caused by a subset $W_1 \subseteq W$, we will now identify the part of the language model used in calculating the probability of W_1. A language model contains many probability distributions and each probability distribution contains many probabilities. One can therefore say that a language model is made up of a set of probabilities $S = \{p_1, ..., p_l\}$. Furthermore, any subset $S1 \subseteq S$ will be called a *part* of the model. In order to calculate the probabilities of a subset W_1 of words (e.g. $p(w_i|w_1, ..., w_{i-1}), i \in I_{W_1}$), the language model will use a subset $S_{W_1} \subseteq S$ of its probabilities. Given a subset W_1, one can then define the part S_{W_1} of the model as the subset of probabilities used to calculate the probabilities of words in W_1. The fraction of LTP caused by a part $S_{W_1} \subseteq S$ of a language model S is then given by f_{W_1}, the fraction of LTP that W_1 causes. This gives the following definition of a weakness of a language model.

Definition: *A part S_{W_1} of a language model S, defined by a subset W_1 of the testing text W, is called a weakness, if W_1 causes a large fraction of LTP.*

The intuitive idea behind this definition is as follows. If subset W_1 causes a large fraction of LTP, then improving it is very important. This conforms to our intuitive meaning of a weakness as something that should be improved in order to improve the overall performance.

This definition is applicable to any probabilistic model that is evaluated in terms of perplexity and that derives a probability for a sequence of tokens by multiplying the probabilities of each token. This includes almost all currently used language models except probabilistic context free grammars and it is also applicable to the language models sometimes used in handwriting or optical character recognition (see [6]). For language models with several components (e.g. class based language models), one can also develop a method (called probability decomposition in [7]), which makes it possible to analyse the weaknesses of the components separately.

One drawback of the given definition of a weakness is that any probability value different from one can potentially be considered a weakness. In other words, a model is compared to a "perfect" model, which would predict every word with a probability of one (even though this is often not possible). Thus, if a language allows for a certain amount of choice, even the best possible model, that would only allow these choices, would still have "weaknesses". In spite of this theoretical drawback, the above definition of a weakness can still be useful from a practical point of view. As one can see from the results given in the next section, it can be used to provide additional insight into a model, potentially leading to an improvement in its performance.

[2] By analogy to TP and LTP, we would prefer to use the term LPP instead of LP. However, since LP is the term commonly used, we will use it as well.

3. ANALYSING WEAKNESSES OF A BI-POS MODEL

We will now apply the definition of a weakness to a commonly used class based bi-gram model, which is also often referred to as bi-pos model. Let $g_1, ..., g_i, ..., g_n$ denote the classes corresponding to the words $W = w_1, ..., w_i, ..., w_n$ in a text [3]. In a bi-pos model, the probability of word w_i is calculated as

$$p(w_i|g_{i-1}) = \sum_{g \in G} p(g|g_{i-1}) * p(w_i|g). \qquad (11)$$

The probabilities $p(a|b)$ are simply estimated from the relative frequencies $f(a|b)$ obtained from a training corpus. However, in order to avoid zero probabilities for events that never occurred, the probabilities can be smoothed with a small constant probability value c_2 (c_1 will be a matching constant to ensure that the sum is equal to one). Furthermore, words that are not part of the vocabulary, so called ***unknown words***, are treated as one unknown symbol, which receives the probability d. The complete formula used in the experiments reported here is deduced from the formula given for a tri-pos model in [5] and it is as follows:

$$p(w_i|g_{i-1}) = \qquad (12)$$
$$\begin{cases} d & \text{if } w_i \notin V \\ (1-d)\sum_{g \in G}[c_1 * f(g|g_{i-1}) + c_2] * f(w_i|g) & \text{else} \end{cases}$$

For more details on the values of the constants and how they are estimated, please refer to [5] or [7].

For our experiments, the training text consists of the first 50,000 words of samples A1-A34 of the Lancaster-Oslo-Bergen (LOB) corpus (see [4]) and the testing text contains roughly 25,000 words from samples A35-A44. As shown in [7], this is sufficient data to train the model. From the set of classes, also called tagset, provided by the LOB, four smaller tagsets were constructed by merging tags with common prefixes (see [7]). All the results reported here, except when indicated otherwise, were obtained with a tagset of 42 tags.

When the model is analysed with respect to its weaknesses, one obtains the following results. First, there is a very small number of tags that accounts for a very large percentage of the LTP (see table 1). A more detailed analysis of these tags, as performed in [8], reveals why these tags cause such a high percentage of *LTP*. Second, depending on the kind of model used for unknown words (see [7]), the fraction of *LTP* caused by unknown words can be as high as 51%. Third, as can be seen from table 2, the prediction of the next word given its tag (e.g. $p(w_i|g)$) is as important as the prediction of the next tag given the previous tag (e.g. $p(g|g_{i-1})$), because it accounts for a large fraction of the *LTP*. These weaknesses, and their more detailed analysis which can not be reproduced here for reasons of brevity, provide us with very useful additional information about the model. For example, one now knows on which contexts one should concentrate ones efforts to improve the model.

Tag	Description	Fraction of *LTP*
N	noun	0.16
AT	article	0.13
IN	preposition	0.12
V	verb	0.08
P	pronoun	0.07
NP	proper noun	0.05
,	comma	0.05
JJ	adjective	0.05
.	period	0.05
BE	forms of to be	0.04

Table 1: The ten tags of the preceding word causing the biggest fraction of the *LTP*

nb of tags	$p(w_i\|g)$	$p(g\|g_{i-1})$	rest
24	0.58	0.35	0.07
42	0.53	0.40	0.07
88	0.45	0.48	0.07
134	0.43	0.50	0.07

Table 2: The *LTP* caused by different components of the model

Given the additional insight from these results, the modeling of unknown words was modified in order to try to overcome the second identified weakness. Rather than having a constant probability of d independent of the context, d is made dependent on the hypothesised tag for the current word. This leads to the following formula:

$$p(w_i|g_{i-1}) = \qquad (13)$$
$$\begin{cases} \sum_{g \in G} d_g * (c_1 * f(g|g_{i-1}) + c_2) & \text{if } w_i \notin V \\ (1-d)\sum_{g \in G}[c_1 * f(g|g_{i-1}) + c_2] * f(w_i|g) & \text{else.} \end{cases}$$

The perplexities of the old model (formula 12) and the new model (formula 13) are shown in table 3. First, we can see that the improvement is substantial for all sets of tags, ranging between 14% and 21%. Second, the improvement increases when the number of tags increases. This is because for each tag, we have a different distribution for unknown words. As the number of tags increases, the distributions of unknown words can become more and more specific.

[3] Even though a word can belong to many classes, each occurrence of a word belongs to only one class.

nb. of tags	old model	new model	improvement
24	265	229	0.14
42	259	218	0.16
88	249	196	0.21
134	243	192	0.21

Table 3: The perplexity of the old and the new model

4. CONCLUSIONS

In this paper, we proposed to analyse the weaknesses of language models in order to subsequently improve the models. A definition of a weakness of a language model that is applicable to most currently used models is given. The application of this definition to a bi-pos model led to the identification of the following three weaknesses: the prediction of the word in a very small number of contexts, the prediction of unknown words and the prediction of the word given its class. These points show that one can obtain additional insight into a model by performing an analysis of its weaknesses. The obtained information is useful in trying to improve the model because one now knows for example on which contexts one should concentrate ones efforts. As an example of this, the modeling of unknown words, which was previously identified as weakness, was changed in order to improve the model. This led to a significant improvement in the performance of the model (up to 21%). Thus, the results presented here show that the given definition of a weakness leads to a better understanding of the model and that the idea of analysing weaknesses of a language model can in general be used to gain considerable insight into a model, potentially leading to a subsequent improvement.

5. REFERENCES

[1] R. O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.

[2] John E. Freund. *Modern Elementary Statistics*. Prentice-Hall, Englewood Cliffs, New Jersey, 7th Edition, 1988.

[3] Fred Jelinek. Self-organized language modeling for speech recognition. In Alex Waibel and Kai-Fu Lee, editors, *Readings in Speech Recognition*, pages 450–506. Morgan Kaufmann, San Mateo, CA, 1990.

[4] S. Johansson, G. Leech, and H. Goodluck. Manual of information to accompany the Lancaster-Oslo-Bergen corpus of British English for use with digital computers. Technical report, Bergen: Norwegian Computing Centre for the Humanities, 1978.

[5] Roland Kuhn and Renato De Mori. A cache-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6):570–583, 1990.

[6] R.M.K. Sinha and B. Prasada. Visual text recognition through contextual processing. *Pattern Recognition*, 21(5):463–479, 1988.

[7] Joerg P. Ueberla. Analysing a simple language model - some general conclusions for language models for speech recognition. *Computer, Speech and Language*, 8:153–176, 1994.

[8] Joerg P. Ueberla. *Analyzing and Improving Statistical Language Models for Speech Recognition.* PhD thesis, School of Computing Science, Simon Fraser University, Burnaby, B.C., V5A 1S6, Canada, May 1994