UNCONSTRAINED HANDWRITING RECOGNITION

Marti, U-V.; Bunke, H.

Published in:
EPRINTS-BOOK-TITLE

Unconstrained Handwriting Recognition: Language Models, Perplexity, and System Performance

U.-V. Marti and H. Bunke
Institut für Informatik und angewandte Mathematik
Universität Bern, Neubrückstrasse 10, CH-3012 Bern
Switzerland
email:{marti,bunke}@iam.unibe.ch

Abstract

In this paper we present a number of language models and their behavior in the recognition of unconstrained handwritten English sentences. We use the perplexity to compare the different models and their prediction power, and relate it to the performance of a recognition system under different language models.

In the recognition experiments a system with the classical architecture of preprocessing, feature extraction and recognition by means of Hidden Markov Model is used. In the recognition phase the language model constrains the possible next words.

Keywords: handwriting recognition, unconstrained English sentence recognition, unigram probability, bigram probability, perplexity.

1 Introduction

In many applications involving natural language processing, language or sentence models are used. There are two main types of language models: the first type is based on a grammar, with strict rules, while the second type rates the words according to probabilities. Other models, such as stochastic context free gramars, can be situated between these two main types.

A simple example of the first type of model is a set of rules describing how basic numeral words can be connected to form higher numeral words. For the other type of language models, many examples can be found in the domain of machine printed OCR, speech recognition [1] but also handwriting recognition [2].

In our work we focus on probabilistic language models in the domain of free handwriting recognition. The system we are building is based on Hidden Markov Models incorporating a language model. The main difference to other systems, for example [3], is that the text is not segmented into words. In our system whole lines of text are treated as basic units. This is motivated by the experiences made in continous speech recognition where the segmentation problem turned out extremely difficult.

In Section 2 we describe the different language models used in our work and in Section 3 how they can be compared to each other. In

In: L.R.B. Schomaker and L.G. Vuurpijl (Eds.), Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition, September 11-13 2000, Amsterdam, ISBN 90-76942-01-3, Nijmegen: International Unipen Foundation, pp 463-468

Section 4 a short overview of the complete recognition system is given. Then in Section 5, the results obtained by comparing different language models with each other are presented. In Section 6 we draw conclusions from this work.

2 Language Models

Let us assume that a text T consists of a sequence $w_1 \dots w_l$ of words w_i out of a vocabulary $V = \{v_i | i = 1 \dots n\}$. From this text T the following numbers can be counted: N - the total number of words in the text T, $N(v_i)$ - the number of occurrences of a word $v_i \in V$ in the text T, and $N(v_i, v_j)$ - the number of occurrences of the word pair (v_i, v_j) in the text T, and so on. Because of limited resources (corpus and computer memory) we did not go further than pairs of words.

In our first model, called "simple sentence model", we assume that all words have the same probability to occur and they are independent of each other. The probability of a word v_i is computed as follows: $p(v_i) = 1/n, \quad \forall v_i \in V$.

In the next model, the "unigram sentence model", the assumption that every word has the same probability to occur, is replaced by the actual occurrence probability $p(v_i)$ of the word v_i, which is determined in the following manner: $p(v_i) = N(v_i)/N, \quad \forall v_i \in V$.

If the actual word influences the choice of the next word, pairs of words have to be regarded. This leads us to the "bigram sentence model" with the probability $p(v_i|v_j) = N(v_j, v_i)/N(v_j), \quad \forall v_i, v_j \in V$.

Because it is possible that a word v_i or a pair of words (v_i, v_j) in the text to be recognized never occur in the training text T, the language model has to be smoothed. This is done by using lower order information instead of higher order one, i.e., bigram probabilities are reduced to unigram probabilities. Therefore the thresholds t_u and t_b give the minimum number of word and word pair occurences. Then the probabilities are computed as follows (N has to be properly adjusted):

$$p(v_i) = \begin{cases} \frac{N(v_i)}{N}, & \text{if } N(v_i) > t_u \\ \frac{t_u}{N}, & \text{if } N(v_i) \leq t_u \end{cases} \tag{1}$$

$$p(v_i|v_j) = \begin{cases} \frac{N(v_j, v_i)}{N(v_j)}, & \text{if } N(v_j, v_i) > t_b \\ b(v_j)p(v_i), & \text{if } N(v_j, v_i) \leq t_b \end{cases} \tag{2}$$

where $b(v_j)$ is a normalization factor to fulfill the probability condition that $p(v_i|v_j)$ summed over j is one (for details see [1]).

3 Perplexity

To compare different language models with each other, test sentences are needed on which one can measure the power of each model. Then, the probability $p(s)$ for a sentence $s = w_1 \dots w_l$ can be computed. One disadvantage of the probability is that it depends on the length of the sentence. To overcome this dependency, the perplexity $\mathbf{P}$ is used:

$$\mathbf{P} = 2^{\mathbf{LP}}; \qquad \mathbf{LP} = -\frac{1}{l} \log_2 \big(p(s)\big) \tag{3}$$

From the point of view of information theory, any language can be seen as an information source. The amount of information from this source is measured by the entropy $\mathbf{H} = -\lim_{l\to\infty} \frac{1}{l} \log_2 p(s)$. This leads to the average branching factor in a graph. So the perplexity $\mathbf{P} = 2^{\mathbf{H}}$ can be seen as the average number of possible successors of a word.

To evaluate the perplexity, sequences s of words are needed. For this purpose we use the Lancaster-Oslo/Bergen (LOB) [4] and the Brown [5] corpus.

When using real text, the language model is incomplete in almost all cases. In particular, there may be words which are not present in the language model vocabulary. A possible method to overcome this problem is to ignore all unknown words or pairs of words where at least one word is unknown. Therefore we set the probabilities $p(\epsilon)$, $p(\epsilon|v_i)$, $p(v_i|\epsilon)$ and $p(\epsilon|\epsilon)$ to one and count the number of unknown words k. If this method is used, the coverage of the text by the vocabulary is given by the value ($c = (l-k)/l$) and the perplexity is:

$$\mathbf{P} = p(s)^{-\frac{1}{l-k}} \tag{4}$$

4 System Overview

In our work we not only aim at investigating theoretical aspects of language models used in handwriting recognition, but we also want to see how these models behave in a recognition system. For this purpose we built a recognition system of unconstrained English text. This system includes three main processing modules: First the handwriting data are preprocessed, then features are extracted from the images of the handwriting, and in the third step we use a Hidden Markov Model for recognition. During Hidden Markov Model training, the language model has no influence; it is only used in the recognition phase. More details of the recognition system can be found in [6].

5 Experiments and Results

In the experiments we wanted to study the usefulness of the different language models for the problem of handwriting recognition. For this

Bigram		Unigram t_u				
t_b		1	10	100	1000	10000
	$\mathbf{P}_{LOB}$	68.8	68.9	70.2	92.4	234.6
	$\mathbf{P}_{Brown}$	67.8	67.9	69.1	90.9	231.5
	$\mathbf{R}$ [%]	76.6	76.6	76.5	76.6	76.6
0	$\mathbf{P}_{LOB}$	21.5	21.5	21.5	21.9	22.3
	$\mathbf{P}_{Brown}$	24.6	24.6	24.6	25.2	25.8
	$\mathbf{R}$ [%]	81.3	81.3	81.3	81.3	81.3
10	$\mathbf{P}_{LOB}$	25.6	25.6	25.5	26.0	27.7
	$\mathbf{P}_{Brown}$	26.4	26.4	26.3	26.9	28.8
	$\mathbf{R}$ [%]	79.8	79.8	79.8	79.9	79.6
100	$\mathbf{P}_{LOB}$	35.8	35.8	35.9	38.7	48.4
	$\mathbf{P}_{Brown}$	35.5	35.5	35.6	38.2	47.5
	$\mathbf{R}$ [%]	78.7	78.6	78.4	78.8	78.7
1000	$\mathbf{P}_{LOB}$	55.9	55.9	56.7	69.2	126.0
	$\mathbf{P}_{Brown}$	55.0	55.0	55.7	67.6	121.6
	$\mathbf{R}$ [%]	78.0	78.0	78.0	78.1	77.8
10000	$\mathbf{P}_{LOB}$	68.8	68.9	70.2	92.4	234.6
	$\mathbf{P}_{Brown}$	67.8	67.9	69.2	90.9	231.5
	$\mathbf{R}$ [%]	77.5	77.3	77.4	77.7	77.8

Table 1: Perplexity $\mathbf{P}$ and recognition rate $\mathbf{R}$ of a 411 word vocabulary system.

purpose we conducted two different sets of experiments. In the first, the vocabulary consists of 411 different words occuring in 541 lines of text written by 6 different writers (set c03-xxx[a-f]) from the database described in [7]. The second, larger system holds a vocabulary of 2346 words in 954 lines written by approximatly 100 different writers (set [a-c]0[1-3]-xxx). In the small system 430 lines were used to train the HMM and 111 lines containing 934 words to test. The large system uses 747 lines for training and 207 lines or 1787 words for testing.

For the simple sentence model, which is our reference model, a word recognition rate of 76.4% for the small vocabulary and 55.1% for the large vocabulary was measured. In this model the perplexity for both vocabularies reaches the maximum: 411 for the small and 2346 for the large vocabulary.

For both systems we have created the smoothed language models (see Sec. 2), using the LOB corpus, and computed the perplexity with the LOB ($\mathbf{P}_{LOB}$) and the Brown ($\mathbf{P}_{Brown}$) corpus. As result, we obtained the values in Table 1 for the small, and in Table 2 for the large system.

The first observation we made is that for increasing smoothing factors the perplexity increases as well. This can be seen for both corpora.

Bigram t_b		Unigram t_u 1	10	100	1000	10000
	$\mathbf{P}_{LOB}$	237.2	238.0	259.6	527.8	1488.9
	$\mathbf{P}_{Brown}$	237.5	236.9	255.6	516.4	1468.1
	$\mathbf{R}$ [%]	55.8	55.8	55.6	55.4	55.2
0	$\mathbf{P}_{LOB}$	51.7	51.7	51.9	53.8	55.3
	$\mathbf{P}_{Brown}$	100.7	100.0	99.9	107.0	113.9
	$\mathbf{R}$ [%]	63.7	63.7	63.7	63.7	63.7
10	$\mathbf{P}_{LOB}$	94.9	94.9	96.1	111.8	135.0
	$\mathbf{P}_{Brown}$	104.5	104.0	104.6	122.7	150.9
	$\mathbf{R}$ [%]	56.0	56.0	56.0	56.0	56.0
100	$\mathbf{P}_{LOB}$	155.7	156.0	163.8	238.9	394.6
	$\mathbf{P}_{Brown}$	158.9	158.2	164.4	237.9	391.4
	$\mathbf{R}$ [%]	56.2	56.1	55.8	55.6	56.0
1000	$\mathbf{P}_{LOB}$	218.1	218.8	236.2	435.9	1011.1
	$\mathbf{P}_{Brown}$	217.6	217.0	231.5	421.8	972.6
	$\mathbf{R}$ [%]	56.1	56.2	55.9	55.6	55.4
10000	$\mathbf{P}_{LOB}$	237.2	238.0	259.6	527.8	1488.9
	$\mathbf{P}_{Brown}$	237.5	236.9	255.6	516.4	1468.1
	$\mathbf{R}$ [%]	56.3	56.2	55.9	55.7	55.5

Table 2: Perplexity $\mathbf{P}$ and recognition rate $\mathbf{R}$ of a 2346 word vocabulary system.

Furthermore we observe for both systems that the values of the bigram models are smaller than those of the unigram models for small smoothing factors. This can be explained by the fact that more information about the language is stored if the smoothing factor is smaller. If the smoothing factor t_b is large, the perplexity is the same as in the unigram model, i.e. the bigram information is lost. If we look at the unigram smoothing factor t_u in the bigram models we see that the perplexity doesn't change much. This means that the unigram probabilities $p(v_i)$ have a rather small influence on the prediction power, while the bigrams are more important.

The comparison between the recognition rate $\mathbf{R}$ (= correct words / tested words) and the perplexity $\mathbf{P}$ shows that the smaller the perplexity, the higher is the recognition rate of the system. In the small system we see that the best recognition rate of 81.3% is achieved by the language model with the smallest perplexity of $\mathbf{P}_{LOB} = 21.5$ ($c_{LOB} = 20\%$) or $\mathbf{P}_{Brown} = 24.6$ ($c_{Brown} = 18\%$). With an increasing smoothing factor the perplexity gets larger and the recognition rate drops down to the minimum. In the larger system the smallest perplexity $\mathbf{P}_{LOB} = 51.7$ ($c_{LOB} = 46\%$) or $\mathbf{P}_{Brown} = 100.7$ ($c_{Brown} = 42\%$) gives the maximum

recognition rate of 63.7%. This drops down to the minimum of 55.4% with perplexity $\mathbf{P}_{LOB} = 1488.9$ or $\mathbf{P}_{Brown} = 1468.1$. It can be seen for both systems that the influence of the bigram smoothing is much larger than of the unigram smoothing.

6 Conclusion

Through this work we have seen that the perplexity of a certain language model can give hints to how good the model behaves in a recognition system. It is difficult to compare different language models using only the perplexity, but it closely related with the performance of the system. If a number of language models have been generated, a fast way to determine the possibly best among them is to calculate the perplexity on a suitable text. Because recognition experiments on large data sets are very time consuming, this may be an interesting first step in testing language models.

1. F. Jelinek. Self-organized language modeling for speech recognition. In A. Waibel and K.-F. Lee, editors, *Readings in Speech Recognition*, pages 450–506. Morgan Kaufmann Publishers, Inc., 1990.
2. S.N. Srihari. From pixel to paragraph: the use of contextual models in text recognition. In *Proc. of the Int. Conf. on Document Analysis and Recognition, Tsukuba Science City, Japan*, pages 416–423, 1993.
3. G. Kim, V. Govindaraju, and S.H. Srihari. Architecture for handwritten text recognition systems. In *Proceedings of Sixth Int. Workshop on Frontiers in Handwriting Recognition 98, Taejon, South Korea*, pages 113–122, 1998.
4. S. Johansson, G.N. Leech, and H. Goodluck. *Manual of Information to accompany the Lancaster-Oslo/Bergen Corpus of British English, for use with digital Computers.* Department of English, University of Oslo, Oslo, 1978.
5. W.N. Francis. *Manual of Information to Accompany a Standard Sample of Present-Day Edited American English for Use with Digital Computers.* Providence, Rhode Island: Department of Linguistics, Brown University.
6. U. Marti and H. Bunke. Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition system. *To appear in Int. Journal of Pattern Recognition and Artificial Intelligence.*
7. U.-V. Marti and H. Bunke. A full English sentence database for off-line handwriting recognition. In *5th Int. Conference on Document Analysis and Recognition 99, Bangalore, India*, pages 705–708, 1999.