Rheinisch Wesfälische Technische Hochschule Aachen
Lehrstuhl für Informatik VI
Prof. Dr.-Ing. Hermann Ney

Speech Recognition and Language Processing in SS 2003

Language Modelling Using Maximum Entropy

Paul Yaw Tawiah

Matriculation number 239 947

July 31, 2003

Supervisors: Nicola Ueffing
Maja Popovic

Language Modelling Using Maximum Entropy

Table of Contents

Abstract

Abstract

This seminar paper discusses language modelling using the principle of Maximum Entropy. Maximum Entropy here is a statistical model which integrates long distance linguistic information with other sources.

Each information source gives a set of constraints to be used on the combined estimate. The intersection of these constraints is the set of probability functions which are consistent with all the information sources. The function with the highest entropy is the Maximum Entropy solution.

An adaptive language model based on maximum entropy approach was trained on Wall Street Journal corpus, and showed a 32-39% perplexity reduction over the baseline. When interfaced to SPHINX-II, Carnegie Mellon's speech recognizer, it reduced its error rate by 10-14%. This thus illustrates the feasibility of incorporating many diverse knowledge sources in a single unified statistical model.

1. Introduction

1.1 Language Modelling

The rationale of language modelling is speech recognition, machine translation and the understanding of natural language. Language modelling is thus an attempt to characterise, capture and exploit regularities in natural language. In statistical language modelling, large amount of text is used to automatically determine the model's parameters in a process known as training which will be discussed in section 2.4 Simply put, a statistical language model describes probabilistically the constraints on word order found in a language. One can view natural language as a stochastic process, in which every sentence, or document or contextual unit of text is treated as a random variable with some probability distribution. For example, a zero order model may recognise that the word "the" occurs much more than any other word. A statistical language model can take advantage of this information when making its predictions.

Let's assume a sentence $w_1^N = w_1, w_2, \ldots, w_N$, where the w_n^i's are the words.
A way to estimate the probability of this sentence $Pr(w_1^N)$ is:

$$Pr(w_1^N) = \prod_{n=1}^{N} Pr(w_n|h_n) \text{ where } h_n = w_1^{n-1} \text{ is called the history.}$$

Language model's performance is measured by the decrease in error rate in the system in which the model is incorporated. A frequently used measure of language model's performance is the perplexity (Bahl et al, section ix) which is based on established information theory principles.

Perplexity is defined as:

$$PP(w_1^N) = \left(Pr(w_1^N)\right)^{-1/N} = \left(\prod_{n=1}^{N} Pr(w_n|h_n)\right)^{-1/N}$$

$$Log\ (PP) = -1/N\ (Log\ Pr(w_1^N)) = -1/N\ \left(\sum_{n=1}^{N} Log\ Pr(w_n|h_n)\right)$$

The next section discusses the conventional language modelling.

1.2 Conventional Language Modelling

The conventional language modelling deals with the n-gram, basically the trigram language model. Actually an n-gram uses the last n-1 words of the history as its sole information source. Thus the trigram predicts a word w_n from (w_{n-2},w_{n-1}) a bigram from w_{n-1} etc. For example, in a simple trigram, the probability of a word w_n, given the previous two words, w_{n-1} and w_{n-2} may be given by:

$$Pr(w_n/w_{n-1}w_{n-2}) = c(w_{n-2}w_{n-1}w_n)/c(w_{n-2}w_{n-1}) \text{.....(1)}$$

where c(x) are counts (number of occurrences of event x).

In reality there is no enough training text to see a large sample of trigrams for any given bigram key, w_{n-1} and w_{n-2}. Considering that all trigrams $w_{n-2}w_{n-1}w_n$ which have not been seen in a training data have a probability zero, the model would have a very poor performance. Instead, bigrams are included in the model to assist cases where the trigram had never been seen. Now because bigram involves only two words instead of three, a large sample of bigrams will be seen in the training data. Similarly there is a need to incorporate unigrams into the model, and the three probabilities are linearly interpolated.

Thus the probability of a word w_n, given previous two words w_{n-1} and w_{n-2} is:

$$Pr(w_n/w_{n-1}w_{n-2}) = \lambda_1 c(w_n)/N + \lambda_2 c(w_{n-1}w_n)/c(w_{n-1}) + \lambda_3 c(w_{n-2}w_{n-1}w_n)/c(w_{n-2}w_{n-1}) \text{ ---(2)}$$

where $\lambda_1 + \lambda_2 + \lambda_3 = 1$

N is the number of words in the training corpus, the c( )'s are the counts of occurrences in the training corpus, λ are parameters optimized over some heldout text not seen during accumulation of the c( )'s. The reason for that is, the heldout text should approximate unseen test data on which the model will ultimately be measured.

2. Maximum Entropy Modelling

2.1 Maximum Entropy Approach

Entropy has many interpretations: it is often used as a measure of randomness and disorder, where high entropy indicates extreme randomness. For classification purposes it can be thought of as a measure of uniformity. Maximum entropy modelling is then a framework for finding the most uniform function which satisfies a number of heterogeneous constraints. The data for a classification problem are described as a number of features. Each feature corresponds to a constraint on the model. The maximum entropy model is then computed, as the model with the maximal entropy of all the models that satisfy the constraints.

One merit of using the maximum entropy approach in language modelling is that the principle provides a well defined method for incorporating different types of dependencies into a single language model. Although there are a number of papers on the application of maximum entropy in natural language processing, a number of open questions and problems in the context of maximum entropy modelling still exist.

- The relation between conventional n-grams model and a maximum entropy model;
- Deficiencies of maximum entropy models in comparison with conventional approach;

2.2 Features and Constraints

The aim is to construct a statistical model of a process under study from the given training sample pairs (x,c), where x is observation and c is the corresponding class. Constraints of such statistical model are defined by so called feature functions. Feature function f(x,c) is a non-negative function of an observation x and a class c. Mostly feature functions are binary functions that return 1 (true) if x belongs to the class c and 0 (false) if not.

The expected value of the feature f with respect to the empirical distribution p(c|x) is exactly the statistic being interested in and it's denoted by:

$$Ep(f) \equiv \sum_{x,c} p(x|c)f(x,c)$$

Now when the empirical probability distribution is assumed to have the same value as the expected value of f in the training sample, then we call such a requirement a constraint equation or simply a constraint. If features are binary valued functions of

(x,c), a constraint means equality of the expected value of the feature function in the model to its expected value in the training data.

2.2.1 Feature Selection

This subsection is concerned with the problem of feature selection: finding the most useful set of features from a large pool of candidates. This process has several benefits; there may be spurious regularities in the data given irrelevant features, creating overfitting (a model overfits if the model has features which are too specific for a given text, and that model cannot be used successfully for the prediction of another unseen text of different context). The principle of maximum entropy does not directly concern itself with the issue of feature selection but provides a recipe for combining constraints into a model.

Consider a large collection F of candidate features. Only a small subset of this will be used in the model.

A small set S is the selected from the full set of candidate features and it is called the *active set.*

This set must be able to capture most of the information about the random process as possible, but should only include features whose expected values can be estimated. An incremental approach to feature selection is used to find the set S. The aim is to build up S by successively adding features and the choice of feature to be added is determined by the training data. If adding a feature f to the set S increases likelihood of training data S becomes SUf, otherwise the feature is discarded.

2.2.2 Model

Defining the Maximum Entropy model;

Consider the training data:

(x_n,c_n) where $x_n \in X_D$ and $n = 1,\ldots,N$.

This training data can be summarized as count table:

$$N(x,c) = \sum_{n=1}^{N} \delta(x,x_n)\delta(c,c_n).$$

Maximum Entropy model is given as:

$$p_\Lambda(c|x) = \exp[\sum_i \lambda_i f_i(x,c)] \,/\, \sum_{c'} \exp[\sum_i \lambda_i f_i(x,c')]$$

where $\Lambda = \{\lambda_i\}$

The normalization constraint for each observation x is:

$$\sum_c p(c|x) = 1$$

The feature constraint for each feature i is :

$$\sum_{x,c} N(x,c).p(c|x).f_i(x,c) = N_i, \text{ where } N_i \text{ are feature counts.}$$

The derivation of the Maximum Entropy Model is thoroughly given in [8].

2.3 Training

2.4.1 Training Data

When one wants to study a random process, the behaviour of the process is observed over a period of time, thus collecting a large number of samples $(x_1,c_1),\ldots,(x_n,c_n)$ where the x_i's are the observations and the c_i's are the classes in that process.

Thus the training sample in terms of its empirical probability distribution p, can be defined as:

$p(x,c) \equiv 1/N$ where N = number of times that the pair (x,c) occurs in the sample

A particular pair (x,c) will either not occur in the training sample or will occur at most a few times. For the case of Language Modelling if one wants to obtain a model for English words, not all the English words can be used to obtain the model. So a sample of English words relating to a particular domain for example sports is taken and used as the basis of obtaining the model. This sample is called the training data.

2.4.2 Generalised Iterative Scaling (GIS)

This is a procedure for finding the parameters $\Lambda = \{\lambda_1 \ldots \lambda_i \ldots \lambda_I\}$ of the maximum entropy distribution $p_\Lambda(c|x)$.

GIS algorithm:

- compute feature counts N_i, thus $N_i = \sum_{x,c} N(x,c).p(c|x).f_i(x,c)$
- choose initial values for $\Lambda = \{\lambda_i\}$
- for each iteration do
  - initialize: $G(\Lambda) = 0$, where $G(\Lambda) = \sum_{x,c} N(x,c).\log p_\Lambda(c|x)$
  - initialize: $Q_i(\Lambda) = 0$ for each feature i, where

$$Q_i(\Lambda) = \sum_{n=1}^{N} \sum_c p_\Lambda(c|x_n) f_i(x_n,c)$$

- for each sample n = 1…N do
  - calculate $Z_\Lambda(x_n)$, where $Z_\Lambda(x_n) = \sum_c \exp[\sum_i \lambda_i f_i(x_n,c)]$

    ($Z_\Lambda(x_n)$ is the normalization factor)
  - accumulate for perplexity:

    $$G(\Lambda) = G(\Lambda) + \log p_\Lambda(c|x_n)$$
  - for each class index c do
    - for each feature $i \in I(x_n,c)$ do
      - accumulate $Q_i(\Lambda)$:

        $$Q_i(\Lambda) = Q_i(\Lambda) + p_\Lambda(c|x_n)$$
- update λ_i for each feature i:

  $$\lambda_i' = \lambda_I + (1/F) \log [N_i / Q_i(\Lambda)]$$

The iteration is continued until convergence.
Generalized Iterative Scaling has the following merits:

- It lends itself to incremental adaptation. New constraints can be added at any time and old constraints can be maintained.
- The GIS algorithm is always guaranteed to converge to the unique Maximum Entropy solution which is guaranteed to exist for consistent constraints.

The demerits of the GIS are

- It is computationally expensive (see Rosenfeld (1994b):section 5.7)
- While the algorithm is guaranteed to converge, there is no theoretical bound on its convergence rate.
- Sometimes it is important to impose constraints that are not satisfied by the training data. This is not good because the constraints may no longer be consistent, and the theoretical results guaranteeing existence, uniqueness and convergence may not hold.

3. Language Modelling and Maximum Entropy

This chapter discusses different information sources which can be used in Language Modelling as well as how Maximum Entropy is used to integrate different information sources.

3.1.1 Information Sources

Conventional n-gram or short term history uses the last n-1 words of the history as it's information source. That means that a bigram predicts w_i from w_{n-1}, a trigram predicts from (w_{n-2},w_{n-1}), etc. The n-gram models are easy to implement, however they have some demerits:

• They are completely "blind" to any phenomenon or constraint which is outside their limited scope.

• The predictors in n-gram models are defined by their ordinal place in the sentence, not by their linguistic role.

Class based n-gram models which increase the reliability of estimates by clustering the words into classes. There are three general methods for word clustering;

1. Clustering by Linguistic Knowledge
2. Clustering by Domain Knowledge
3. Data Driven Clustering

A more detailed paper on clustering can be gotten from Rosenfeld (1994b).

Long distance bigrams use the word w_{n-d} as history, where d is the distance.
The amount of information in long distance bigrams was estimated in Huang et al (1993a) and was found that long distance n-grams are seriously deficient in that, although they capture word-sequence correlations even when the sequences are separated by a distance d, they fail to appropriately merge training instances that are based on different values of d.
The table below shows the training perplexity of long distance bigrams for various distances, based on 1 million words of the Brown Corpus. The distance = 1000 was included as a control.
For long distant bigrams the conditional probability is denoted by $Pr(w_n|h_n)$, where $h_n = w_{n-d}$ and d = distance.

Distance	1	2	3	4	5	6	7	8	9	10	1000
PP	83	119	124	135	139	138	138	139	139	139	141

As expected, the perplexity was found to be low for d = 1, and increased significantly through d = 2,3 ,4 and 5. But For d = 5,…,10, the training-set perplexity remained at about the same. It was then concluded that significant information existed only in the last four words of the history. It was concluded that information does indeed exist in the more distant past, but it lightly spread across the entire history.

Trigger pairs:

A→B is considered a trigger pair if the word sequence of A is significantly correlated with another word sequence B. A is said to be the trigger and B the triggered sequence.

A study was conducted on the Wall Street Journal corpus of 38 million words in attempt to design a trigger-based model. The following conclusions were derived at the end:

1. Different trigger pairs display different behaviour, and hence should be modelled differently.
2. Self triggers are particularly powerful and robust .That means, for more than two third of the words, the highest mutual-information trigger proved to be the word itself.
3. When the trigger and the triggered word are from different domains of discourse, the trigger pair actually shows poor mutual information.

3.2 Combining Information Sources: Linear Interpolation and Backingoff

This section deals with combining the different estimates from the different knowledge sources to arrive at an optimal estimate to be included in a model. Linear interpolation and Backing-off are some of the solutions that tackle the problem and are briefly described in this subsection.

Linear interpolation is a method that can be used as a way of combining knowledge sources.

Linear combination of k models $\{P_i(w|h)\}_{i=1\ldots k}$;

$$P_{COMBINED}(w/h) \overset{def}{=} \sum_{i=1}^{k} \lambda_i P_i(w|h), \text{ where } o \leq \lambda_i \leq 1 \text{ and } \sum_i \lambda_i = 1.$$

The Estimation Maximization algorithm is used to determine the weights λ_i (see Dempster et al., 1977 for the algorithm).

Linear interpolation has some merits, which make it the method used in many situations:

- Because of its generality, any language model can be used as a component.
- Linear interpolation is easy to implement, experiment with, and analyse. There is an interpolated program that takes any number of probability streams, and an optimal bi-partitioning stream, and runs the EM algorithm to convergence (see Rosenfeld,1994b: appendix B) and its been concluded that :
  - The exact value of the weights does not significantly affect perplexity.
  - Very little heldout data are enough to arrive at reasonable weights.
- It is assumed that the interpolated model is not worse than any of its individual components.

Apart from the merits of linear interpolated model it also has some demerits:

- Linearly interpolated model make suboptimal use of its components since all the individual components have their strengths and weaknesses. This means that the combined model does not make optimal use of the information at its disposal.
- Linearly interpolated models are generally inconsistent with their components. Thus each information source typically partitions the event space (h,w) and provides estimates based on the relative frequency of training data within each class of the partition. Therefore within each of the component models, the estimates are consistent with the marginal of the training data, but this reasonable measure of consistency is in general violated by the interpolated model.

Backoff method; here the different information sources are ranked in order of detail and specificity, and the model with the most detailed information is consulted first at runtime. If it is found to contain enough information, this model is then used to generate the estimate, else the next model is consulted. Differently from the linear interpolation, backingoff does not combine all the information sources but chooses some among them to generate the estimate.

Backoff method has merits of being simple, compact and often better than linear interpolation and a demerit of exhibiting discontinuity around the point where the decision is made of choosing the best model with most detail and specifity.

A common problem of linear interpolation and backingoff is that they give rise to systematic overestimation of some events (see Rosenfeld and Huang (1992) and Chase et al.(1994) for a discussion of this problem).

3.3 Maximum Entropy in Language Modelling

This subsection discusses how the Maximum Entropy principle is used for language modelling.The conventional formulations of n-grams, i.e the unigram, bigram and trigram are replaced by their respective feature functions conveying the same information.

Feature function for the unigram w_n:

$$f_{w_n}(h,w) = \begin{cases} 1 \text{ if } w = w_n \\ 0 \text{ otherwise} \end{cases}$$

The empirical expectation in the training data is given by the formula below:

$$\bar{E}p[f_{wn}] = 1/N \left[\sum_{h,w} f_{wn}(h,w)\right], \text{ where (h,w) belongs to the training data.}$$

The respective associated constraint is also:

$$\sum_{h} þ(h) \sum_{w} p(w|h) f_{wn}(h,w) = \bar{E}[f_{wn}] \quad \text{.....(II)}$$

where þ denotes the empirical distribution.

Feature function for the bigram:

$$f_{\{w1,w2\}}(h,w) = \begin{cases} 1 \text{ if h ends in } w_{n-1} \text{ and } w = w_n \\ 0 \quad \text{otherwise} \end{cases}$$

Feature function for the trigram:

$$f_{\{wn-1,wn-2,wn\}}(h,w) = \begin{cases} 1 \text{ if h ends in } (w_{n-1},w_{n-2}) \text{ and } w = w_n \\ 0 \quad \text{otherwise} \end{cases}$$

The respective associated constraints for the bigram and trigram are evaluated in a similar way as that of the unigram shown above by simply replacing their respective constraint functions in the equation (II).

Feature function for trigger pair A→B:

$$f_{A \blacktriangleright B} = \begin{cases} 1 \quad \text{if A belongs to the history and w = B} \\ 0 \quad \text{otherwise} \end{cases}$$

Assume that for each word w in a vocabulary V a set of n_w trigger words $t_{w1},t_{w2}\ldots t_{wnw}$ is defined. The relative frequency of a trigger $t \in \{ t_{w1},t_{w2}\ldots t_{wnw} \}$ and a word w is given by:

$$d(t,w) = c(\text{t belongs to h and w immediately follows h})/N$$

Feature function for long distant bigram is:

$$f(t,w) = \begin{cases} 1 \text{ if t belongs to h but h does not end with t and w is the next word after h} \\ 0 \text{ otherwise} \end{cases}$$

Maximum entropy model for Language Modelling:

$P_\Lambda(c|x) \overset{LM}{\Rightarrow} P_\Lambda(w_n|h_n)$, so the model for the conditional probability of the language model becomes:

$$P_\Lambda(w_n|h_n) = \exp[\sum_i \lambda_i f_i(h_n,w_n)] \,/\, \sum_{w_n'} \exp[\sum_i \lambda_i f_i(h_n,w_n')].$$

Normalization constraint: $\sum_{w_n} P_\Lambda(w_n|h_n) = 1$

The corresponding feature constraint for each feature i is:

$$\sum_{h,n} N(h_n,w_n) P_\Lambda(w_n|h_n).f_i(h_n,w_n) = N_i$$

3.4 Assessing Maximum Entropy Approach

The last subsection of this chapter points out the merit of Maximum Entropy modelling in comparison with Linear Interpolation and Backingoff methods.

- The Maximum Entropy principle is simple and it imposes all of the constituent constraints, but assumes nothing else.
- It is very general, thus many other knowledge sources can be incorporated and constraints do not have to be independent or uncorrelated with each other.
- Any information captured by existing language models can be absorbed into the Maximum Entropy model.

4. Adaptation in Language Modelling

The adaptation in language modelling grew out from the idea to improve the conventional trigram language model by extracting information from the document's history. This approach is also called "long distance modelling". The trigram model is rather naïve, because by intuition, the likelihood of word's appearance depends on more than the previous two words. For example, one would reasonably expect that the context of the text has a large impact on the probability of a word's appearance; if looking at a news story about stock market, the probability of financial terms should be increased. Much of this information is not captured by the trigram model. Also, the trigram model is static in that it does not take into account the style or patterns of word usage in the source text. For example, in a speech recognition system, the speaker may not match the given limited vocabulary. The trigram model is not able to adapt to the speaker's vocabulary and style, nor to topic information which is contained in a document. An improvement upon the trigram model was proposed in Kuhn and Demori [3] and Jelinek et al. [4]. The cache model takes into account the relative dynamic frequencies of words in text seen so far and incorporates this information into the prediction maintaining a cache of recently used words. This results up to 23% improvement in perplexity as reported in Jelinek et al.[4]. However the manner in which the cache component was combined with the trigram component was a simplistic linear interpolation which was described in section section 3.3.1. In Della Pietra et al [5], a more disciplined approach was pursued. The static trigram model was considered as the baseline probability distribution, the dynamic cache distribution was then imposed as a constraint on the probability distribution. The final probability was the one which satisfied the cache constraint. Improvement of up to 28% in perplexity was reported, using a small test set of 321 words.

Another method in which the trigram model can be improved is to take triggers, or related words into account. A trigger pair, A→B as described in section 3.1, is a word pair such that the appearance of A in the current document history significantly affects the probability of seeing B. For example, in the Wall Street Journal corpus the presence of the word "brokers" in a document makes the appearance of the word "stock" approximately five times more than normal. The additional presence of the word "investors" increase the probability of "stock" by another 53% (Lau et al[1], Rosenfeld and Huang [6]. The subsection below now discusses the paradigms of adaptation.

4.1 Paradigms of Adaptation

There are three main paradigms of adaptation;

- Within-domain adaptation
- Cross-domain adaptation
- Limited-data domain adaptation

In the *within-domain adaptation*, a heterogeneous language source is treated as a complex product of multiple domains-of-discourse ("sublanguages"). The goal here is then to produce a continuously modified model that tracks sublanguage mixtures, sublanguage shifts etc.
Maximum entropy models are naturally suited for within domain adaptation, although they can be modified for cross-domain adaptation as well. This is because constraints are derived from the training data.

Cross-domain adaptation paradigm is one in which the test data comes from a source to which the language model has never been exposed. Thus, the training data and test data are assumed to come from different sources. When this happens, the result is a significant degradation in language modelling quality. The further apart the two language sources are, the larger the degradation.

With the *limited-data domain adaptation* paradigm, moderate amounts of training data are available from the test domain. Larger amounts of data may be available from other, "outside", domains. This situation is often encountered in real-world applications. For detailed discussion on the paradigms of adaptation in language modelling see R.Rosenfeld, 1996 [7].

4.2 Maximum Entropy for Adaptive Statistical Language Modelling

To maximize adaptive performance in statistical language modelling, the maximum entropy model was supplemented with the unigram and bigram caches (the cache model is described above). The conventional trigram was used as the baseline and was especially important for the 38MW system, since it employed a high cuttoffs on n-gram constraints. The table below summarizes perplexity (PP) performance of various combinations of the trigram model, the Maximum Entropy model, and the unigram and bigram caches:

Vocabulary ……Top 20 000 words of WSJ corpus
Test set……325KW

Training set	1MW		5MW		38MW	
	PP	%change	PP	%change	PP	%change
Trigram (baseline)	269	-	173	-	105	-
Trigram + Caches	193	-28	133	-23	88	-17
Maximum Entropy	203	-24	123	-29	86	-18
ME + Trigram	191	-29	118	-32	75	-28
ME+Trigram+Caches	163	-39	108	-38	71	-32

The following results and analysis from the table above were summarised as follows:

- The trigram had a static perplexity and was used as the baseline.
- Trigram + Cache: This experiment represents the best adaptation achievable without the ME formalism, because the amount of information provided by the caches is independent of the amount of training data and therefore a relative improvement provided by caches is greater.
- Maximum Entropy: The relative advantage of "pure" Maximum Entropy models seems to be greater with more training data. This is because ME uses constraint functions to capture correlations in the training data. The more data, the more statistically reliable n-gram and trigger correlations exist, and the more constraints are employed.
- ME + Trigram: When the Maximum Entropy model is interpolated with conventional trigram, the most significant perplexity reduction occurs in the

38MW system. This is because the 38MW ME model employed high n-grams cuttoffs, and was thus "blind" to the low count n-gram event.

- ME + Trigram + Caches: This was the best adaptive scheme that was achieved (R. Rosenfeld). Compared with trigram + cache experiment, the addition of ME component improves perplexity by a relative 16% for the 1MW system, and by a relative 19% for the 5MW and 38MW systems.

5. Conclusion

The goal of this seminar paper was discussing Language Modelling using the principle of Maximum Entropy.

The Maximum Entropy Modelling:

- Provides the most uniform probability distribution which satisfies a number of different constraints.
- Possibility of incorporating many different information sources.
- Any information obtained by existing language models can be integrated into Maximum Entropy model.
    1. Reformulate the different estimates as constraints on the expectation of various functions, to be satisfied by the target (combined) estimate.
    2. Among all probability distributions that satisfy these constraints choose the one that has the highest entropy.

Other methods for finding estimates from different information sources were discussed, that is the linear interpolation and the backingoff methods were discussed in chapter 3.3 and it was found that the Maximum Entropy method chooses an optimal estimate that satisfies all the constraints form the different information sources, whereas the other two methods do not.

The static nature of the conventional language modelling lead to the discussion on adaptation in statistical language modelling: the idea is to improve the conventional trigram language model using information from document's history. The Maximum Entropy principle lead to maximal performance in statistical language modelling when interpolated with trigram and cache models, thus reducing the perplexity by 32% on a vocabulary of 20 000 words of the Wall Street Journal corpus with a test set of 325KW and a training set of 38MW.

Acknowledgement

I am very grateful to Nicola Ueffing and Maja Popovic for many suggestions and discussions.

6. References

[1] Lau,.R, Rosenfeld,R., and Roukos, S., "Adaptive Language Modelling using Maximum Entropy Approach," in ARPA Human Language Technology Workshop, March 1993

[2] Lau, R., Maximum Likelihood Maximum Entropy Trigger Language Model, S.B Thesis, MIT, May 1993

[3] Kuhn,R. and De Mori,R., " A Cache-Based Natural Language Model for speech recognition," in IEEE Transaction on Pattern Analysis and Machine Intelligence, 12(6): 570-583, 1990

[4] Jelinek, F., Merialdo,B., Roukos, S. and Straus, M., "A Dynamic Language Modelling for Speech Recognition", in proceedings of the Speech and Natural Language DARPA workshop, 293-295, February 1991

[59] Della Pietra, S. and Della Pietra,V., Personal Communications, 1993

[6] Rosenfeld, R. and Huang,., "Improvements in Stochastic Language Modelling," in proceedings of the Speech and Natural Language DARPA workshop, Morgan Kaufmann Publishers, San Mateo CA, February 1992

[7] Ronald Rosenfeld, "A Maximum Entropy Approach to Adaptive Statistical Language Modelling", Computer Speech and Language, Vol. 10, No. 3, July 1996.

[8] Herman Ney "Maximum Entropy and Log-Linear Modelling" draft 10-Oct-01