CONSTRUCTION OF LANGUAGE MODELS USING THE MORPHIC GENERATOR GRAMMATICAL INFERENCE (MGGI) METHODOLOGY

E. Segarra[*] , *L. Hurtado*
Dept. Sistemas Informáticos y Computación
Universidad Politécnica de Valencia (Spain)
E-mail: esegarra,lhurtado@dsic.upv.es, Tel.: 34 6 3877738

ABSTRACT

Over the last few years, some alternatives to N-gram language models, which are based on stochastic regular grammars, have been proposed. These grammars are estimated from data through Grammatical Inference algorithms. In particular, the Morphic Generator Grammatical Inference (MGGI) methodology has been applied to tasks of written natural language queries to databases. As for N-gram models, language models obtained through this methodology require the use of smoothing techniques.

This work incorporates a version of the well-known Back-Off smoothing method to the MGGI language models to solve the estimation problem of unseen events in the training corpus, and shows the behaviour of the smoothed MGGI models in two tasks of written sentences. The results illustrate that the smoothed MGGI model works better than the standard smoothed bigram model.

1. INTRODUCTION

Statistical language models are extensively used in Automatic Speech Recognition tasks. Trigrams, bigrams, and triPOS, etc. have been compared, combined and interpolated to improve the recognition results ([1],[2],[3])]. The main advantage of the N-gram approach consists in its ability to learn parameters automatically. N-grams are based on the estimation of the probability of observing a given linguistic unit, which is conditioned on the observation of N-1 preceding linguistic units, and the number of probabilities to be taken into account is an exponential function of N. Therefore, a large training corpus is needed for a correct estimation of such a great number of probabilities. Thus, in practice, the use of models of this kind is reduced to low values of N, bigrams and trigrams, and, consequently, the information supplied by these models is local.

In recent years, some alternatives to N-grams, which are based on formal grammars ([4],[5]), have been proposed. They propose the application of regular inference algorithms MGGI [6] and ECGI (Error Correcting Grammatical Inference) [7] to language modeling. These alternatives incorporate the main attractive features of the N-gram approach: that is to say the models are estimated from corpora and they are based on regular grammars allowing for a simple integration of the language model with acoustic models. On the other hand, they use general regular grammars which allow for a representation of the language structure without the locality constraints of the N-grams.

Both N-gram and grammatical models obtained through regular inference have to solve the problem of the estimation of the probabilities of events which are not represented in the training corpus, i.e. unseen events. In the N-gram approach, this problem has been extensively discussed resulting in some well-known smoothing methods: linear and non-linear interpolation ([8],[1]), back-off smoothing [9], etc. The back-off method is the most widely used and adapted method because of its good results ([2],[10],[11]).

This work incorporates a version of the back-off smoothing method to the MGGI language models in order to solve the estimation problem of unseen events in the training corpus and shows the behaviour of the smoothed MGGI models in two tasks of written sentences. Preliminary experimentation has been presented in [12].

In Section 2, a survey of the MGGI methodology is presented. In Section 3, the adaptation of the back-off smoothing method to MGGI models is presented. In section 4, the behaviour of the smoothed MGGI models in two tasks of written sentences is illustrated, and in Section 5, some conclusions are presented.

2. THE MGGI METHODOLOGY

The MGGI methodology [6] was introduced as a step towards a general methodology for inference of Regular Languages. Its definition methodology is based on the following two points: a) the generative property of 2-Testable in the Strict Sense Languages (2-TSSL) [13] allows us to obtain arbitrary Regular Languages by applying morphic operators to 2-TSSL (the concept of stochastic 2-TSSL coincides with the concept of bigram); and b) the main drawback of the 2-TSSL inference method [13] is that, if we attempt to use it directly as a Grammatical Inference procedure, it will generally lead to "overgeneralized" languages.

[*] This work is partially supported by the Spanish CICYT under grant TIC95-0884-C04-01, and by the Spanish PIITA-UPV-60960206/279.

Let *R* be a sample over the alphabet *Σ*. Let *Σ'* be a finite alphabet. Let *h* be a letter-to-letter morphism, *h: Σ'*→Σ**, and *g* a renaming function, *g: R→Σ'**.

The Regular Language, *L*, generated by the MGGI-inferred grammar, *G*, is related to *R* through the expression: *L = h(l(g(R)))*, where *l(g(R))* is inferred from *g(R)* through the 2-TSSL inference algorithm [13].

Example:

Let *Σ*={a,b} be an alphabet and let *R*={aaba, abba, abbbba, aabbbbaa} be a sample over *Σ*. Two different grammatical inference algorithms are used:

a) *2-Testable in the Strict Sense inference algorithm (its stochastic version is equivalent to bigrams):*

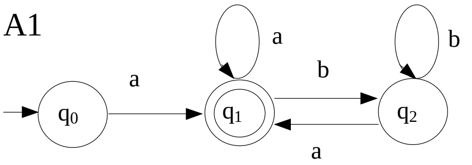

Figure 1. The finite automaton inferred from *R* by the 2-Testable in the Strict Sense algorithm.

The language accepted by the automaton of Figure 1 is: L(A1) = a + a(b+a)*a. State q_1 represents history *a*. State q_2 represents history *b*.

b) *MGGI algorithm*: *(non-stochastic version)*

The renaming function *g: Σ*→Σ'** is the **relative position** with 2 intervals:
g(R) = {$a_1a_1b_2a_2$, $a_1b_1b_2a_2$, $a_1b_1b_1b_2b_2a_2$, $a_1a_1b_1b_1b_2b_2a_2a_2$}

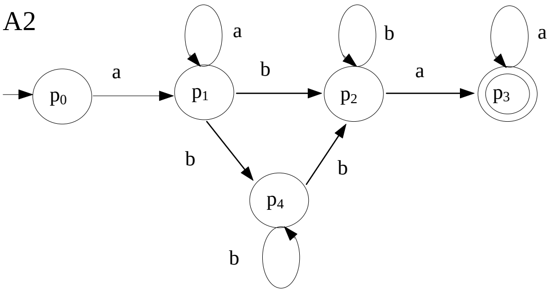

Figure 2. The finite automaton inferred from *R* by MGGI algorithm, with a relative position renaming function.

The language accepted by the automaton of Figure 2 is: L(A2) = $a^+b^+a^+$. State p_1 represents history a_1. State p_2 represents history b_2. State p_3 represents history a_2. State p_4 represents history b_1. If we apply the inverse morphism to these histories, then states p_1 and p_3 represent history *a*, and states p_2 and p_4 represent history *b*. Therefore, there is a relation between the states of automata A1 and A2: p_0 with q_0, p_1 and p_3 with q_1, p_2 and p_4 with q_2. If the related states of automaton A2 are clustered then the derived automaton is equivalent to A1.

The relation between the states in the MGGI automaton and the 2-Testable in the Strict Sense automaton led us to a strategy for smoothing the stochastic version of the MGGI models: the use of the bigram model as the lower-level probability distribution.

On the other hand, there is another interesting feature of the MGGI methodology. Given a task (a language to modelize), we can choose an adequate definition of a renaming function *g*. Different definitions of this function produce different models.

3. BACK-OFF IN AN MGGI MODEL

Let $P_S\left(w_n \middle| w_1^{n-1}\right)$ be the smoothed probability of the n-gram w_1^n, and let *r* the number of times that this n-gram has been seen in the training text. The back-off method [9] defines this probability as follows:

$$\begin{cases} P_s(w_n|w_1^{n-1}) = d_r P_{ML}\left(w_n \middle| w_1^{n-1}\right) & r > 0 \\ P_s(w_n|w_1^{n-1}) = \alpha(w_1^{n-1}) P_s(w_n|w_2^{n-1}) & r = 0 \end{cases}$$

where d_r is 1 for values of *r* greater than a given threshold *k*.

In MGGI models we smooth in a local manner. For each state *q* of the model, we calculate the probability of a word *w* when the model is in state *q*, that is *P(w|q)*. In the smoothing process of MGGI models, the lower-level probability distributions used to smooth are:

a) Labeled unigram model

The probability distribution of the unigram model inferred from the labeled sample *g(R)* is the lower-level probability distribution used for smoothing. The process is the same as in an N-gram model.

b) Unlabeled bigram model

The probability distribution of the bigram model inferred from *R* is the lower-level probability distribution used for smoothing. We use (P_S*) to denote the probability of the unlabeled bigram model**.**

Let *q* be a state of the model whose associated history *w'* is such that *h(w')=*w_{n-1}. Therefore, the state *q* is related to the state in the unlabeled bigram model which represents the history w_{n-1}. The conditioned probability of word w_n, which has never been seen in state *q*, is denoted by the expression:

$$P_S\left(w_n \middle| q\right) = \alpha(q) P_S^*(w_n|w_{n-1})$$

where α is:

$$\alpha(q) = \frac{1 - \sum_{w_n : r>0} P_s(w_n|q)}{1 - \sum_{w_n : r>0} P_S^*(w_n|w_{n-1})}$$

4. EXPERIMENTAL RESULTS

In order to evaluate this technique, we did a series of experiments with two different corpora. The first corpus consisted of written sentences (1,178 different words) of queries to a Spanish geography database [14]. The average number of words per sentence was 9.9. To train the model, we used 8,000 sentences of the corpus. To test it, we used 1,246 different sentences which were vocabulary-dependent.

The renaming function *g* of the MGGI methodology defined in experiments was the relative position (see example in Section 2), with values from 2 to 9 for the number of intervals.

In order to estimate the threshold value *k* of the back-off scheme, we did a series of experiments in which the first 7,000 sentences of the training set were used to train the different models. The following 947 sentences of the same set were used as a test set, and we calculated its perplexity. We explored values for *k* from 2 to 17. The lower-level probability distribution used for smoothing was the labeled unigram model.

LM	N. of units	Max. K	Average Perpl.	% Impr.
Bigram	1,120	12	12.4323	-
MGGI-2	1,585	9	11.2818	9.25
MGGI-3	1,924	13	10.8393	12.81
MGGI-4	2,220	8	10.6661	14.21
MGGI-5	2,499	17	10.6541	14.30
MGGI-6	2,764	15	10.6644	14.22
MGGI-7	3,017	11	10.6445	14.38
MGGI-8	3,238	12	10.6877	14.03
MGGI-9	3,459	13	10.7247	13.73

Table 1. The table shows the following for each language model: the number of units of the models, the maximum value allowed for the threshold *k*, the average test set perplexity and the percentage of its improvement with respect to smoothed bigram model.

In Table 1, the average results for each estimated model (bigrams, and MGGI models with a number of intervals from 2 to 9) are shown. The MGGI model with a relative position renaming function of 5 intervals (MGGI-5) gave a 14.30% improvement with respect to the smoothed bigram model, and gave a number of units of 2,499. A number of intervals of 7 (MGGI-7) gave a slightly greater improvement (14.38%) than the above model, but the number of units increased to 3,017. From this series of experiments, we defined the best MGGI model as having a relative position renaming function of 5 intervals and a threshold *k* of 12, which gave the best results for the model of 5 intervals.

MG GI	Smoothing prob. distr.	k bigram	Perplexity
5	labeled unigrams	---	10.1831
5	unlabeled bigrams	9	10.1484

Table 2. Test set perplexity for the MGGI language model for the two lower-level probability distributions used for smoothing the model.

Finally we trained this best model with the complete training corpus of 8,000 sentences (1,178 different words) and we calculated the perplexity for the 1,246 sentences of the test set. The results obtained are shown in Table 2. The number of units (states) of the language model was 2,647.

The test set perplexity of the back-off smoothed bigram model was 11.7287 (with 9 for the value of the threshold). Therefore the MGGI model gave a 13.18% improvement with respect to the bigram model when the model was smoothed with the labeled unigram model, and an improvement of 13.47% when the model was smoothed with the unlabeled bigram model.

The second corpus consisted of 500,000 written sentences taken from a tourist guidebook with 689 different words [15]. The average length of the sentences was approximately 9.9. From these 500,000 sentences we took all the different sentences which defined a set of 172,283 sentences. We divided this set into two parts: the first 140,000 sentences constituted the training set, and the last 32,283 constituted the test set, which was vocabulary-dependent.

The renaming function *g* of the MGGI methodology defined in experiments was also the relative position, with values from 2 to 9 for the number of intervals.

As in the other task, in order to estimate the threshold value *k* of the back-off scheme, we also did a series of experiments. The first 90,000 sentences of the training set were used to train the different models. The following 10,000 sentences of the same set were used as a test set, and we calculated its perplexity. We explored values for *k* from 2 to 15. The lower-level probability distribution used for smoothing was the labeled unigram model.

LM	N. of units	Max. K	Average Perpl.	% Impr.
Bigram	671	7	7.9170	-
MGGI-2	904	8	6.8435	13.56
MGGI-3	1,230	15	6.5607	17.13
MGGI-4	1,571	15	6.3546	19.73
MGGI-5	1,895	15	6.2852	20.61
MGGI-6	2,064	12	6.2311	21.29
MGGI-7	2,416	15	6.2196	21.44
MGGI-8	2,636	14	6.2107	21.55
MGGI-9	2,935	15	6.2400	21.18

Table 3. The table shows the following for each language model: the number of units of the models, the maximum value allowed for the threshold *k*, the average test set perplexity and the percentage of its improvement with respect to smoothed bigram model.

In Table 3 the average results for each estimated model (bigrams, and MGGI models with a number of intervals from 2 to 9) are shown. The MGGI model with a relative position renaming function of 6 intervals (MGGI-6) gave a 21.29% improvement with respect to the smoothed bigram model. The MGGI-6 model gave a number of units of 2,064. A number of intervals of 8 (MGGI-8) gave a slightly greater improvement (21.55%) than the above model, but the number of units increased to 2,636. From this series of experiments, we defined the best MGGI model as having a relative position renaming function of 6 intervals and a threshold *k* of 11, which gave the best results for the model of 5 intervals.

We trained this best model with the complete training corpus of 140,000 sentences (680 different words). We calculated the test set perplexity for the 32,283 sentences. The results obtained are shown in Table 4. The number of units (states) of the language model was 2,097.

MGGI	Smoothing prob. dist.	k bigram	Perplexity
6	labeled unigrams	---	6.2310
6	unlabeled bigrams	5	6.1822

Table 4. Test set perplexity for the MGGI language model for the two lower-level probability distributions used for smoothing the model.

The test set perplexity of the back-off smoothed bigram model was 7.8454 (with 5 for the value of the threshold). Therefore, the MGGI model gave an improvement of 20.58% with respect to bigram model when the model was smoothed with the labeled unigram model, and an improvement of 21.2% when the model was smoothed with the unlabeled bigram model.

5. CONCLUSIONS

The results obtained confirm the appropriateness of the smoothing method we have chosen to be incorporated to the regular inference algorithm MGGI. Both the labeled unigram model and the unlabeled bigram model used for smoothing the MGGI language models work adequately, and they gave a similar performance.

On the other hand, the results obtained show that all the MGGI language models work better than the smoothed bigram model, in spite of the simplicity of the renaming function *g* defined in this work. It is worth noting that every regular language can be inferred with this methodology (trigrams for example), and its potential modeling power is encouraging.

6. REFERENCES

[1] H.Ney, U.Essen. *"On Smoothing Techniques for Bigram-Based Natural Language Modeling"*. ICASSP'91, pp 825-828, 1991.

[2] P.Placeway, R.Schwartz, P.Fung, L.Nguyen. *"The Estimation of Powerful Language Models from Small and Large Corpora"*. ICASSP'93, pp II-33/II-36,1993.

[3] M.Generet, H.Ney, F.Wessel. *"Extensions of Absolute Discounting for Language Modeling"*. EUROSPEECH'95, pp 1245-1248, 1995.

[4] E.Segarra, P.García. *"Automatic Learning of Acoustic and Syntactic-Semantic Levels in Continuous Speech Understanding"*. EUROSPEECH'91, pp 861-864, 1991.

[5] N.Prieto, E.Vidal. *"Learning Language Models through the ECGI Method"*. Speech Communications, N 1, pp 299-309, 1992.

[6] P.García, E.Segarra, E.Vidal, I.Galiano. *"On the Use of the Morphic Generator Grammatical Inference (MGGI) Methodology in Automatic Speech Recognition"*. International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), Vol 4, N 4, pp 667-685, 1990.

[7] H.Rulot, N.Prieto, E.Vidal. *"Learning Accurate finite-state Structural Models of Words through the ECGI algorithm"*. ICASSP'89, pp 643-646, 1989.

[8] F.Jelinek. *"Markov Source Modeling of Text Generation"*. NATO-ASI In the Impact of Processing Techniques on Communications. Martinus Nijhoff Eds., pp 569-598, 1985.

[9] S.Katz. *"Estimation of Probabilities From Sparse Data for the Language Model Component of a Speech Recognizer"*. IEEE Trans. on ASSP, Vol 34, N 3, pp 400-401, 1987.

[10] G.Bordel, I.Torres, E.Vidal. *"Back-off Smoothing in a Syntactic Approach to Language Modeling"*. ICSLP'94, pp 851-854, 1994.

[11] G.Ricardi, E.Bocchieri, R.Pieraccini. *"Non Deterministic Stochastic Language Models for Speech Recognition"*. ICASSP'95, pp 237-240, 1995.

[12] E. Segarra, L. Hurtado. *"Application of the Back-Off Smoothing Technique to the MGGI Language Models"*. VII National Simposium on Pattern Recognition and Image Analysis. Vol II, pp 34-35, 1997.

[13] P.García, E.Vidal. *"Inference of k-Testable Languages in the Strict Sense and Application to Syntactic Pattern Recognition"*. IEEE Trans. on PAMI, Vol 12, N 9, pp 920-925, 1990.

[14] J.Díaz, A.Rubio, A.Peinado, E.Segarra, N.Prieto, F.Casacuberta. *"Development of Task-Oriented Spanish Speech Corpora"*. EUROSPEECH'93, 1993.

[15] J.C.Amengual et al. *"First-Phase Final Report"*. ESPRIT-IV <LTR>:EUTRANS, 1996.