Statistical Feature Language Model

Kamel Smaïli, Salma Jamoussi, David Langlois, Jean-Paul Haton

Statistical Feature Language Model

Kamel Smaïli, Salma Jamoussi, David Langlois and Jean-Paul Haton

INRIA-LORIA, Speech Group
B.P. 101 - 54602 Villers les Nancy, France
Tel.: +33 (0)3 83 59 20 83 - Fax: +33 (0)3 83 27 83 19
e-mail: smaili,jamoussi,langlois, jph@loria.fr - http://www.loria.fr/equipes/parole

Abstract

Statistical language models are widely used in automatic speech recognition in order to constrain the decoding of a sentence. Most of these models derive from the classical n-gram paradigm. However, the production of a word depends on a large set of linguistic features : lexical, syntactic, semantic, etc. Moreover, in some natural languages the gender and number of the left context affect the production of the next word. Therefore, it seems attractive to design a language model based on a variety of word features. We present in this paper a new statistical language model, called Statistical Feature Language Model, SFLM, based on this idea. In SFLM a word is considered as an array of linguistic features, and the model is defined in a way similar to the n-gram model. Experiments carried out for French and shown an improvement in terms of perplexity and predicted words.

1. Introduction

Statistical language models are widely used in speech recognition to constraint the decoding process to choose at each step the best words. They are based on the prior probability of a linguistic unit (often word) given a history of the same type (typically n-grams). Other linguistics units can be taken into account by integrating them in a specific model (typically n-classes). In this case, separate models are constructed and their output are combined. Another method, more interesting but more complex, is based on the maximum entropy [1] and integrates in the same framework the features which come from each model. The chosen model satisfying all the constraints is the one with the highest entropy. Other methods contribute to the improvement of perplexity by making the context larger and more significant without increasing the complexity [2]. One of the problem of statistical language models is to consider the word depending on only precedent words or classes. Whereas, in natural language the production of a word depends on several features: lexical, syntactic, semantic, ... In fact and particularly in French for instance, the gender and number of the left context of a word affect the production of the next word. In the evaluation of a speech recognition system, if the word *mangé* has been recognized instead of *mangées* [1], the system considers it as an error. In order to take into account a maximum number of word features, we propose in this paper a new statistical method based on features (SFLM).

2. Statistical Feature Language Model

In inflected natural languages like French, linguistic features are very useful to reduce speech recognition errors due to homonyms. Therefore, we propose a Statistical Feature Language Model in which a word is viewed as an array of m features, so that:

$$W = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_m \end{pmatrix}$$

Each f_i is a linguistic characteristic of W. These characteristics or features could be the word itself, its syntactic class, its gender, its number, its semantic class, ...
The classical n-gram model is defined by:

$$P(w_1 \dots w_n) = \prod_{i=1}^{n} P(w_i | w_{i-1} \dots w_{i-k+1}) \qquad (1)$$

By analogy, we define a SFLM by:

$$P(w_1^{1\dots m}, \dots w_n^{1\dots m}) = \prod_{i=1}^{n} P\left(\begin{pmatrix} f_1^i \\ f_2^i \\ \vdots \\ f_m^i \end{pmatrix} \middle| \begin{pmatrix} f_1^{i-1} \\ f_2^{i-1} \\ \vdots \\ f_m^{i-1} \end{pmatrix} \dots \begin{pmatrix} f_1^{i-k+1} \\ f_2^{i-k+1} \\ \vdots \\ f_m^{i-k+1} \end{pmatrix} \right) \qquad (2)$$

where $w_i^{1\dots m}$ is the feature array corresponding to the i^th word and $(f_1^j, f_2^j, \dots, f_m^j)^t$ indicates the j^{th} feature array word of the history. Bilmes [3] proposes a similar model supported by a graphical model. The model we propose here is very simple to implement with the

[1] *mangé* and *mangées* have the same pronunciation

classical language modeling toolkits (CMU, SLRI). In fact, what we propose is to replace each word in the training and test corpora by its feature array and let a LM toolkit to run a SFLM.

3. SFLM in practice

The implementation of SFLM consists of assigning to each word its m features. In order to validate our approach in a real case, in the following experiments, we decided to use a SFLM of two features. We choose as first feature the word itself, and its syntactic class as second feature.

3.1. Syntactic classes

To syntactically cluster the vocabulary, we use the following rule : A word is put in a class if any other word of the same class can be substituted to it in a context without any change in the syntactic structure of the sentence. To make that possible, we defined some syntactic contexts and checked all the words of the dictionary on these contexts. In this clustering a word can belong to several classes. This leads to a set of 230 classes which have been used to tag all the corpora with the Viterbi algorithm [4].

3.2. Corpora and vocabulary

The training and test corpus have been extracted from "Le Monde" newspaper. We used 40 million words for training and 1,8 words for tests. Both corpora have been labelled as described before. For the vocabulary, we selected all the words occurred more than 20 times. This leads to a vocabulary of 47000 units.

4. An Overview of the Shannon's Game

The Shannon game [5] has been adapted in [6] in order to give an alternative method to perplexity for evaluating language models. The aim of this new evaluation protocol is to estimate the prediction capacity of a language model. This protocol has been used in a comparative evaluation campaign for language models organized by AUPELF-UREF in which we have taken part [7].

A set of truncated sentences is provided to the model and is used as a test corpus. The goal of this protocol consists of supplying a list of candidate words for each truncated sentence. To each word is associated a bet which estimates the likelihood of the candidate word. To do that, based on the history (the truncated sentence), the language model bets on each vocabulary word by assigning a value between 0 and 1. Therefore, the perplexity is evaluated as the inverse of the geometric mean of the bets placed on the correct words. To control the volume of data, the number of candidates for each truncated sentence has been limited to the top list. All the candidates are sorted in decreasing bet. If the correct word is not in the ordered list of candidates, its probability is set to a floor value.

In the experiments we did, we have investigated:

- the number of times the word to guess is proposed at the first rank,
- the number of times the word to guess is proposed in the five first ranks,
- the mean rank of words to guess calculated over all the truncated sentences,
- the Shannon perplexity,
- the percentage of correct words retrieved in the top list, over all the truncated sentences.

In our experiments, we randomly truncated each sentence of the test corpus. This leads to about 57500 truncated sentences (and as many words has to be guessed). We fixed the size of the top list to 5000 [6].

5. Results and comments

SFLM has been tested in terms of perplexity and Shannon's game on large corpora and has been compared to baseline models. The classical perplexity has been obtained with CMU toolkit. The SFLM we used is based on two features but can be obviously extended to several other features.
In order to make the comparison relevant, we decide to test the models with the same vocabulary and the same training and test corpora. Each word in the training corpus C_{word} has been replaced by its features vector (FV) which leads to a new corpus $C_{features}$. Therefore, from $C_{features}$ we extract a vocabulary $V_{features}$ of about 47000 FV (each of them occurred more than 20 times). Then, with this material we run a 2-gram, a 3-gram and a 4-gram models with several discounting methods as shown in table 1. For the baseline models we construct a vocabulary V_{words} from $V_{features}$ by eliminating all the features of a word and by keeping only one occurrence of each word.
Some interesting points are brought to light by this table that are worth mentioning in passing. For all n-grams, SFLMs outperform the corresponding baseline methods when the perplexity is considered without UNK. The improvement reaches 7% for 2-grams, 6% for 3-grams and $4,8\%$ for 4-grams. On the other hand, this observation is not true when we take into account UNK. This does not constitute a drawback of our approach because language models have to be evaluated in terms of perplexity by excluding UNK. The UNK has an important probability due to the sparseness data and this makes the perplexity decreasing abnormally.

Another important remark concerns the perplexity obtained by the trigram SFLM (105.65) which is equivalent or more exactly slightly better than a classical 4-gram (105.94). This illustrates the importance of integrating in statistical language model the linguistic characteristics of words.
In order to investigate deeply this approach and before testing it in a real speech recognition system, we tested its capability to predict words. For that we did several experiments in the framework of Shannon's game. In this experiment we use a vocabulary of 20K feature word and a bigram a trigram language models. The truncated corpus (57500 sentences) is the the one we used in the experiments described in table 1.
Table 2 shows that, by using SFLM more than 200 words have been recognized in addition in the first rank and more than 400 words in the first five ranks. This result is very important for a speech recognition system, it shows that it will be possible to improve the word error by recovering more correct words. Even if the percentage of correct words has not changed, the mean rank of recognized words has been improved by 8 points by using a bigram FLMS.

6. Some tracks of improvement

FLMs leads to increase the size of the vocabulary, therefore the data sparseness problem increases, and as for classical n-gram models, we have to develop an efficient backoff strategy. We can backoff as usual to the smaller model by considering the feature word as a single block. This what we did in this work, but we can backoff more efficiently by using a smaller word array feature. In order to highlight our matter, let us to give an example. If a word feature array $(f_1, \ldots, f_i, \ldots, f_m)$ is not seen in the training corpus, then the vector $(*, \ldots, f_i, \ldots, f_m)$ or $(f_1, \ldots, *, \ldots, f_m)$, etc. may have more chance to be in the training corpus. This could lead to the notion of parallel backoff graph as proposed by Bilmes [3]. There are several ways to backoff and only few paths are interesting. For each history, across the backoff graph, we have to find the best path(s). This issue is very important to benefit of the whole potential of the approach we proposed. In such a complex set of word features, we must, for each history, find towards which features we do backoff. In precedent work, we developed a method named the Selected History Principle [8, 9] which can partially gives a solution to this problem. This principle allows to measure the prediction capacity of a language model (and therefore the best backoff strategy) for one history. This principle will be the cornerstone of our backoff strategy.

7. Conclusion

In this paper, we present a new approach for statistical language modeling. This approach considers that linguistic units are complex objects including several kind of informations: orthographic form, syntactic features (gender, number, possible syntactic role in the sentence), semantic features (related topic, paradigmatic relationships with others 'words'...). Classically, one develops a different language model dedicated to each kind of features and then models are combined. We argue that all features should be involved in the same model. The response we give consists in integrating them into lexical units by creating Feature Vectors.

We present first experiments on SFLM by using only two features. The results show that this new approach is capable to outperform the classical models in terms of perplexity and predictable words. In terms of perplexity, a trigram SFLM model improves the perplexity of a classical trigram model by 6% and reaches the performance of a classical 4-gram model, when unknown words are not included. The Shannon game provides a better oracle for future integration in speech recognition. Our approach allows to rank more words at rank 1 than classical n-gram models. This is promising for speech recognition because, at final, only the 1-best output is provided by the speech decoding system.

We think improving this approach by investigating three directions:

- First, we will add new features coming from our precedent works: topic features [10], semantic concepts [11], gender, number...
- Second, we will estimate the benefit of including continuous valued features. The estimation of parameters will have to be adapted, but in a first step, it will be possible to go back to the discrete case by using quantization.
- Third, as explained in previous section, we will develop a strategy based on the Selected History Principle, in order to find efficient backoff strategies among backoff graph [3].

The good results we obtained encourage us to introduce this model in our speech recognition system based on Julius open source [12] as part of the first step-pass decoding process. This will be done easily due to the facility of using features in our approach. The current vocabulary of our speech recognition system will be transformed by replacing each word by its feature vector.

8. References

[1] R. Rosenfeld, "Adaptive statistical language modeling: A maximum entropy approach," Ph.D. dissertation, School of Computer Science Carnegie

		Good Turing		Linear		Witten Bell	
		With UNK	Without UNK	With UNK	Without UNK	With UNK	Without UNK
2-gram	Without features	146.02	173.71	151.95	180.82	147.40	174.43
	With features	153.49	160.24	160.31	167.37	155.03	161.67
3-gram	Without features	96.80	112.24	115.01	133.28	98.71	113.68
	With features	101.92	**105.65**	122.0	126.42	104.09	107.69
4-gram	Without features	92.18	105.94	119.77	137.54	96.02	110.15
	With features	97.44	**100.75**	126.92	131.19	101.68	105.01

Table 1: Comparison of SFLM and baseline models performance in terms of perplexity with and without unknown word

Absolute							
			Rank of retrieved words		**Other measures**		
			1	≤ 5	*Mean Rank*	*Shannon PP*	*% correct word rate*
2-gram	Without features	With UNK	9275	20792	188.67	141.05	97.9
		Without UNK	8218	18511	197.66	156.95	93.38
	With features	With UNK	9479	21220	180.36	129.22	97.9
		Without UNK	8353	18727	189.59	144.92	93.0
3-gram	Without features	With UNK	12148	24487	156.91	94.99	97.96
		Without UNK	10990	22347	164.32	104.23	93.43
	With features	With UNK	12384	24837	152.27	87.63	97.95
		Without UNK	11128	22467	160.01	96.85	93.11

Table 2: Comparison of SFLM and baseline models performance in terms of perplexity with and without unknown word

Mellon University, Pittsburgh, PA 15213, April 1994.

[2] I. Zitouni, K. Smaili, and J.-P. Haton, "Statistical language modeling based on variable-length sequences," *Computer Speech and Language*, vol. 17, no. 1, pp. 27–41, Jan 2003.

[3] J. Bilmes and K. Kirchhoff, "Factored language models and generalized parallel backoff," in *Proceeding of Human Language Technology Conference*, Edmonton, Canada, 2003.

[4] K. Smaïli, A. Brun, I. Zitouni, and J. Haton, "Automatic and manual clustering for large vocabulary speech re cognition: A comparative study," in *European Conference on Speech Communication and Technology*, Budapest, Hungary, September 1999.

[5] C. Shannon, "Prediction and entropy of printed english," *Bell System Technical Journal*, vol. 30, pp. 50–64, 1951.

[6] F. Bimbot, M. El-Bèze, S. Igounet, M. Jardino, K. Smaïli, and I. Zitouni, "An alternative scheme for perplexity estimation and its assessment for the evaluation of language models," *Computer Speech and Language*, vol. 15, pp. 1–13, 2001.

[7] M. Jardino, F. Bimbot, S. Igounet, K. Smaïli, I. Z. touni, and M. El-Beze, "A first evaluation campaign for language models," in *First International Conference on Language Resources and Evalu ation*, Granada, Spain, May 1998, pp. 801–805.

[8] D. Langlois, K. Smaïli, and J.-P. Haton, "Clustering words contexts based on statistical language models prediction capability," in *Third International Conference on Modeling and Using Context*, Dundee, Scotland, July 2001.

[9] D. Langlois, K. Smaïli, and J.-P. Haton, "Retrieving phrases by selecting the history: application to automatic speech recognition," in *7th International Conference on Spoken Language Processing - ICSLP'2002*, vol. 1, Denver, Colorado, 2002, pp. 721–724.

[10] A. Brun, K. Smaïli, and J. Haton, "Contribution to topic identification by using word similarity," in *International Conference on Spoken Language Processing (ICSLP2002)*, 2002.

[11] S. Jamoussi, K. Smaili, and J.-P. Haton, "Understanding process for speech recognition," in *Eighth European Conference on Speech Communication and Technology - EuroSpeech'03, Genève, Suisse*, Sep 2003.

[12] A. Lee, T. Kawahara, and K. Shikano, "Julius – an open source real-time large vocabulary recognition engine," in *EUROSPEECH'2001*, 2001, pp. 1691–1694.