Arabic statistical language modeling

Karima Meftouh[1], Kamel Smaili[2], Mohamed-Tayeb Laskri[1]

[1]Badji Mokhtar University – Computer Science Department – BP 12 23000 Annaba – Algeria

[2]INRIA-LORIA – Parole team – BP 101 54602 Villers Les Nancy – France

Abstract

In this study we propose to investigate statistical language models for Arabic. Several experiments using different smoothing techniques have been carried out on a small corpus extracted from a daily newspaper. The sparseness of the data leads us to investigate other solutions without increasing the size of the corpus. A word segmentation technique has been employed in order to increase the statistical viability of the corpus. This leads to a better performance in terms of normalized perplexity.

Keywords: Arabic language, statistical language model, text corpora, perplexity, segmentation.

1. Introduction

A statistical language model is used to build up sequence of words, classes or phrases which are linguistically valid without any use of external knowledge. A list of probabilities is estimated from a large corpus to indicate the likelihood of linguistic events. An event is any potential succession of words. This kind of models is useful in a large variety of research areas (Goodman, 2001): speech recognition, optical character recognition, machine translation, spelling correction… The common model used in the literature is the well known n-grams. A word is estimated in accordance to the $(n-1)$ previous words. To be efficient these models need a huge amount of data to train all the needed parameters.

For Arabic, the necessary resources are not as important as what we have for the Indo-European languages. This is due to the relative recent interest for Arabic applications. In this paper, we investigate several classical statistical language models in order to study their pertinence for Arabic language. Sparseness data conducts us to test several smoothing techniques in order to find out the best model.

In the following section, we will give an overview of Arabic language (section 2), we pursue by a description of n-gram models (section 3) then n-morpheme models are presented. Finally we point out different results and conclude.

2. The Arabic language

Arabic, one of the six official languages of the United Nations, is the mother tongue of 300 millions people (Egyptian demographic center, 2000). Unlike Latin-based alphabets, the orientation of writing in Arabic is from right to left. The Arabic alphabet consists of 28 letters and can be extended to ninety by additional shapes, marks and vowels. Each letter can appear in up to four different shapes, depending on whether it occurs at the beginning, in the middle, at the end of a word, or alone. Table 1 shows an example of the letter ف / “f” in its various forms. Letters are mostly connected and there is no capitalization.

Isolated	Beginning	Middle	End
ف	فـ	ـفـ	ـف

Table1: The letter ف */ "f" in its various forms*

Arabic is a Semitic language. The grammatical system of Arabic language is based on a root-and-pattern structure and considered as a root-based language with not more than 10000 roots and 900 patterns (Hayder and al., 2005). The root is the bare verb form. It is commonly three or four letters and rarely five. Pattern can be thought of as template adhering to well-known rules.

Arabic words are divided into nouns, verbs and particles. Nouns and verbs are derived from roots by applying templates to the roots to generate stems and then introducing prefixes and suffixes (Darwish, 2002). Table 2 lists some templates (patterns) to generate stems from roots with examples from the root درس/ "drs":

Template	Stem
فعل CCC	درس "drs"/ Study
فاعل CACC	دارس "dArs"/ Student
مفعول mCCwC	مدروس "mdrws"/ Studied

Table2: Some templates to generate stems from the root درس/ *"drs"*
C indicate a consonant, A a vowel.

Many instances of prefixes and suffixes correspond to a word in English, such as pronouns and propositions. An example of an Arabic word is given in Table 3: وكررتها "and she repeats it".

Arabic contains three genders (much like English): masculine, feminine and neuter. It differs from Indo-European languages in that it contains three numbers instead of the common two numbers (singular and plural). The third one is the *dual* that is used for describing the action of two people.

Arabic	English
و	and
كرر	repeats
ت	she
ها	it

Table3: An example of an Arabic word

3. N-gram models

The goal of a language model is to determine the probability of a word sequence w_1^n, $P(w_1^n)$. This probability is decomposed as follows:

$$P(w_1^n) = \prod_{i=1}^{n} P(w_i / w_1^{i-1}) \quad (1)$$

The most widely-used language models are n-gram models (Stanley and Goodman, 1998). In n-gram language models, we condition the probability of a word on the identity of the last $(n-1)$ words.

$$P(w_i / w_1^{i-1}) = P(w_i / w_{i+1-n}^{i-1}) \quad (2)$$

The choice of n is based on a trade-off between detail and reliability, and will be dependent on the available quantity of training data (Stanley and Goodman, 1998).

4. N-morpheme models

Languages with rich morphology generate so many representations from the same root. Often, this makes them highly flexional and consequently the perplexity could be important (Kirchoff and al., 2002). An Arabic word consists of a sequence of morphemes respecting the following pattern *prefix*-stem-suffix** (* denotes zero or more occurrences of a morpheme). We define an n-morpheme model as an n-gram of morphemes. In this case the corpus is rewritten in terms of morphemes as in the example of figure 1.

سوق_اهراس
انجراف ال تربة يستهلك ال ملايير دون جدوى عجزت ال مصالح ال تقنية
لـ ولاية سوق_اهراس عن مجابهة مشكل انجراف ال تربة ال ذي يلحق
أضرارا كبيرة بـ شتى ال قطاعات و تنجر عنه خسائر مالية هامة تكون
على حساب متطلبات تنموية أخرى و هو ما جعل ال ولاية من أكثر ال
ولايات تخلفا و أكثرها تأخرا في ال مرافق ال إجتماعية ال ضرورية

Figure1: A sample of Arabic Morphemes corpus

The quality of a language model is estimated by the test perplexity PP:

$$PP = 2^{-\frac{1}{n}\log_2(P(w))} \quad (3)$$

With n is the size of the test corpus.

When we proceed to a decomposition of words into prefix*-stem-suffix*, we modify the number of items constituting the original corpus W. To make the comparison of the two models relevant, the perplexity has to be normalized (Gauvain and al., 1996) as follows:

$$PP_n = 2^{\frac{n_1}{n_2}\log_2(PP)} \quad (4)$$

where n_1, n_2 correspond respectively to the size of the original corpus and the rewritten one.

5. Data description and experimental results

The experiments reported bellow are conducted on corpora extracted from Al-khabar (an Algerian Daily newspaper). Al-Khabar is written on modern standard Arabic, the one used by all the official media in Arabic world. One of the specifity of Arabic language is that a text can be read without using any vowel. That is why articles in newspapers are unvocalized. The corpora we use contain 80K words for training and 5K words for test. Figure 2 shows a sample of the training corpus.

سوق اهراس
انجراف التربة يستهلك الملايير دون جدوى عجزت المصالح التقنية لولاية سوق اهراس عن مجابهة مشكل انجراف التربة الذي يلحق أضرارا كبيرة بشتى القطاعات وتنجر عنه خسائر مالية هامة تكون على حساب متطلبات تنموية أخرى وهو ما جعل الولاية من أكثر الولايات تخلفا وأكثرها تأخرا في المرافق الإجتماعية الضرورية

Figure2: A sample of the training corpus

For the both following experiments the language models have been smoothed by three techniques: Good-Turing (Katz, 1987), Witten-Bell (Witten and Bell, 1991) and linear (Ney and al., 1994).

5.1. Word-based n-gram models

The baseline model is calculated with a vocabulary of the most frequent 2000 words. Table 4 and table 5 show the performance in terms of test perplexity with and without UNK. The rate of unknown words is 30.19%.

Linear		Witten-bell		Good-Turing		n
E	P	E	P	E	P	
8.27	309.29	8.07	267.86	8.18	289.10	2
8.33	321.50	8.12	278.87	8.19	292.36	3
8.39	335.14	8.29	311.97	8.26	307.51	4

Table4: Perplexity and Entropy performance without UNK

Linear		Witten-bell		Good-Turing		n
E	P	E	P	E	P	
6.37	82.92	6.25	76.03	6.26	76.66	2
6.52	92.09	6.35	81.18	6.35	81.55	3
6.61	97.67	6.48	89.25	6.46	88.07	4

Table5: Perplexity and Entropy performance with UNK

Note that the values of perplexity are high and increase according to the order of the model n. This is due to the weak size of the training corpus. To take into account the sparseness data issue, we propose to split words into morphemes. This operation leads to increase the frequency of basic units and consequently to reduce the percentage of unknown words.

5.2. Morpheme-based n-gram models

A word is rewritten by separating its prefixes from the other morphemes. The prefixes which are used for the segmentation are the common used in Arabic language (Table6).

prefixes			
"w" و	and	"l" لـ	to
"k" كـ	like	"b" بـ	with
"f" فـ	then	"Al" الـ	the

Table6: Prefixes and their meanings

To make the corpus statistically reliable and to fit the reality of the Arabic language, some words have been gathered. That is why for instance, we concatenate the town's name composed by two or more words (Zitouni and Smaili, 2000). See Table 7 for an example. This operation is handled by using a predefined list of composed words. Work is under progress to find out automatically sequence of Arabic words.

سوق أهراس Is rewritten: سوق_ أهراس

Table7: An example of composed town's name

The transformation of the initial corpora leads to respectively a training and a test corpus of 110K and 6,9K tokens. Table 8 and table 9 show the values of the unnormalized perplexity with and without UNK.

Linear		**Witten-Bell**		**Good-Turing**		***n***
E	P	E	P	E	P	
6.56	94.25	6.43	86.44	6.46	87.89	2
6.24	75.50	6.03	65.44	6.10	68.42	3
6.25	76.08	6.06	66.70	6.13	69.82	4

Table8: Perplexity and Entropy performance without UNK.

Linear		Witten-Bell		Good-Turing		*n*
E	P	E	P	E	P	
5.95	61.91	5.85	57.66	5.85	57.63	2
5.72	52.81	5.53	46.17	5.56	47.27	3
5.75	53.96	5.56	47.11	5.61	48.72	4

Table9: Perplexity and Entropy performance with UNK.

We remark that 3-gram and 4-gram models lead to better results than bigram. This is due to the fact that this segmentation makes the corpus statistically viable. Indeed, the decomposition decreases the variety of bigrams and increase the frequency of tree and four grams. In order to compare the perplexity to that obtained with the original language models, we compute the normalized perplexity. Table 10 lists these values.

n	Good turing	Witten bell	Linear
2	173.52	170.18	188.05
3	130.05	123.55	145.61
4	133.06	126.23	146.93

Table10: Normalized Perplexity's values

These results show an improvement of 55.7% in terms of 3-gram perplexity using Witten-Bell smoothing technique. We can state that for small corpus, the segmentation of words improve the language model and this, whatever the used technique of smoothing.

5.3. Influence of parameters on perplexity values

In the last experiment we kept the same dictionary as in the baseline model to make the comparison possible. In the following experiments, we investigate two other vocabularies.

- In this experiment, we use a dictionary composed of the best 25% words constituting the corpus. This leads to a vocabulary of 2,4K tokens. The perplexity's values reported in Table 11 are computed on the segmented test corpus of 6959 tokens for which 14.27% are out of vocabulary. Table 12 shows the perplexity's values including UNKs.

Linear		Witten-Bell		Good-Turing		*n*
E	P	E	P	E	P	
6.69	103.08	6.57	94.72	6.58	95.99	2
6.36	82.33	6.17	71.86	6.22	74.64	3
6.37	82.76	6.20	73.44	6.25	76.30	4

Table11: Perplexity and Entropy performance without UNK.

Linear		Witten-Bell		Good-Turing		*n*
E	P	E	P	E	P	
6.16	71.72	6.06	66.70	6.06	66.56	2
5.92	60.70	5.73	53.25	5.77	54.41	3
5.95	61.69	5.77	54.41	5.81	55.94	4

Table12: Perplexity and Entropy performance with UNK

- Now, we consider words occurring 10 or more times to be included in the vocabulary. We obtain a vocabulary size of 1241 words. Table 13 shows the perplexities: 1458 words (20.95%) out of vocabulary. Perplexities including UNKs are given in table 14.

Linear		Witten-Bell		Good-Turing		*N*
E	P	E	P	E	P	
6.14	70.46	6.03	65.49	6.05	66.25	2
5.86	58.11	5.65	50.09	5.71	52.28	3
5.90	59.52	5.68	51.18	5.74	53.40	4

Table13: Perplexity and Entropy performance without UNK

Linear		Witten-Bell		Good-Turing		*N*
E	P	E	P	E	P	
5.33	40.10	5.25	38.10	5.24	37.78	2
5.16	35.68	4.97	31.37	4.99	31.82	3
5.22	37.39	5.01	32.27	5.05	33.05	4

Table14: Perplexity and Entropy performance with UNK

6. Conclusion and future work

In this work we used n-grams to modelize Arabic language; several experiments have been carried out on a small corpus extracted from a daily newspaper. The sparseness data conducts us to investigate other solutions without increasing the size of the corpus. We think that even with a large corpus, segmentation is necessary. In fact, a lot of words in Arabic are constructed from patterns which are used as generative rules. Each pattern indicates not only how to construct a word but gives the syntactic role of the generated word.

Several experiments and developments are under work, the objective is to obtain a very robust statistical Arabic language model. Among them, we can mention:

- A tool capable of segmenting a word into *prefix*-stem-suffix**.
- An evaluation of a complete morpheme-based n-gram model.

- Arabic n-class model.
- A dynamic Bayesian Network formalism for Arabic statistical modeling.

References

Hayder K. Al Ameed, Shaikha O. Al Ketbi and al. (2005). Arabic light stemmer: A new enhanced approach. *Proc. of IIT'05 (the Second International Conference on Innovations in Information Technology)*.

Ghaoui A., Yvon F., Mokbel C. and Chollet G. (2004). Modèle de langage statistique à base de classes morphologiques. *Le traitement automatique de l'arabe, JEP-TALN*, Fès.

Lee Y., Papineni K., Roukos S., Emam O., Hassan H. (2003). Language model based Arabic word segmentation. *Proc. of the 41st Annual meeting of the association for computational linguistics*. 399-406.

Stolcke A. (2002). SRILM, An extensible language modelling toolkit. *Proc. Intl. conf spoken language processing.*

Kirchoff K. et al. (2002). Novel approaches to Arabic speech recognition. Technical report. *Final report from the 2002 Johns-Hopkins summer workshop*. John-Hopkins university.

Darwish K. (2002). Building a shallow Arabic morphological analyser in one day. *Proc. of the ACL workshop on computational approaches to Semitic languages*.

Goodman Joshua T. (2001). A bit of progress in language modeling, extended version. Technical report MSR-TR-2001-72.

Zitouni I. et Smaili K. (2000). Vers une meilleure modélisation du langage : la prise en compte des séquences dans les modèles statistiques. *Journées d'études sur la parole*, Aussois, France. 293-296.

Egyptian Demographic center. (2000). http://www.frcu.eun.eg/www/homepage/cdc/cdc.htm.

Stanley F. Chen, Goodman J. (1998). An empirical study of smoothing techniques for language modelling. Technical report TR-10-98, Computer science group, Harvard University, Cambridge, Massachusetts.

Clarkson P. et Rosenfeld R. (1997). Statistical language modelling using the CMU-Cambridge Toolkit. *Proc. Eurospeech,* Rhodes, Greece.

Gauvain J.L., Lamel L., Adda G, and Matrouf D. (1996). The LIMSI 1995 Hub system. *Proc. ARPA Spoken Language Technology Workshop-96*.

Ney H., Essen U. and Kneser R. (1994). On structuring probabilistic dependencies in stochastic language modeling. *Computer Speech and Language*, 8(1): 1-38.

Witten I.T. and Bell T.C. (1991). The Zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory*, 37(4): 1085-1094.

Katz S.M. (1987). Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal processing*, 35(3): 400-401.