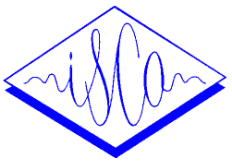

6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

ON ENHANCING KATZ-SMOOTHING BASED BACK-OFF LANGUAGE MODEL

Jian Wu and Fang Zheng

Center of Speech Technology, State Key Laboratory of Intelligent Technology and Systems,
Department of Computer Science and Technology, Tsinghua University, Beijing, China
jwu@sp.cs.tsinghua.edu.cn http://sp.cs.tsinghua.edu.cn

ABSTRACT

Though the statistical language modeling plays an important role in speech recognition, there are still problems that are difficult to be solved such as the sparseness of training data. Generally, two kinds of smoothing approaches, namely the back-off model and the interpolated model, have been proposed to solve the problem of the impreciseness of language models caused by the sparseness of training data. By expanding the idea of interpolation model to Katz-smoothing based re-estimation of the seen word pairs, a back-off model based modified method is proposed, referred to as the enhanced Katz smoothing with deleted interpolation (EKSWDI). A uniform expression and two simplified versions for this modified model are also given. Experiments on a Chinese pinyin-to-character conversion system and the perplexity measure show that the proposed model has a better performance than the Katz smoothing method does.

1. BACKGROUND

Statistical language models are commonly used in the large-vocabulary speech recognition systems. The most frequently used one is the Markov model [1], where the word sequence is considered as the observation of an (N-1)-order Markov process called N-Gram. Under this assumption, the probability of a word sequence $T = (W_1 W_2 \Lambda\ W_n)$ can be approximated by $P(T) = \prod_{i=1}^{n} P\left(W_i \middle| W_{i-N+1}^{i-1}\right)$, where only the most recent (N-1) words are considered to predicate the coming word. In spite of the simplicity and applicability of this formula, the most reasonable Maximal Likelihood Estimation (MLE) used to train the language model has fallen across many difficulties. Firstly, the huge corpus needed for more accurate n-gram estimation is difficult to collect, label, and process. Secondly, no matter how widely the training data covers, the real system should still face the severe data-sparseness problem because there won't be enough domain-specific data available. In order to deal with the above problems, many smoothing methods are proposed. For a traditional tri-gram model (where N=3), the simplest approach is the linear interpolation among the uni-gram, bi-gram and tri-gram probabilities [2]. In this implementation, the total training data are divided into two distinct portions. *Kept* data, the larger one, is used to estimate the conditional probabilities of the focused words given the corresponding historical word sequences while *held-out* data, the smaller one, is used to estimate the weights among the three relative frequencies. Another prevalent method applied in the state-of-the-art speech recognizers is the back-off algorithm [3]. Both the back-off smoothing algorithm and the deleted-interpolation algorithm can generally yield a good performance, but they perform differently when the training data are different in size. The back-off re-estimation is more accurate for large training corpus, while the interpolation re-estimation for small one. Part of our work here is to study the factors that cause the difference and to find approaches to the reconcilably use of them, and therefore an enhanced Katz smoothing based language model integrated with deleted-interpolation is proposed in the paper.

This paper begins with a review and an analysis on both the standard Good-Turing (GT) estimation method and the Katz Smoothing (KS) method in Section 2. In section 3, the detailed algorithm of our proposed integrated model is described. In Section 4 we introduce the experimental setup and describe the perplexity and recognition results. Conclusion is drawn in Section 5. For the ease of explanation, all of the algorithms described in this paper are based on the bi-gram model, which can be extended to a higher order model easily.

2. GOOD-TURING ESTIMATION AND KATZ SMOOTHING

The basic idea of Good-Turing re-estimation is to partition n-grams according to their frequencies so that the n-grams' parameter space can be shared. In the GT estimation, the frequency (also known as the occurring *count*) of any seen n-gram is discounted according to some transcendental rules. Moreover, the accumulated residual probability is re-distributed to the unseen n-grams. The standard rule is presented as follows,

$$r^* = (r+1)\frac{n_{r+1}}{n_r}, \tag{1}$$

where r is the frequency of certain n-gram, r^* is the frequency after discounted and n_r is the number of n-grams that occur exactly r times in the training data. After the normalization, the relative frequency

$$P_{GT}(W_r) = \frac{r^*}{\sum_{r=0}^{\infty} n_r r^*} \tag{2}$$

is taken as the re-estimated probability. The GT re-estimation can assign an approximate value to unseen n-grams (that is to say, n-grams with zero probabilities) to prevent candidates containing

such n-grams from being neglected out of consideration. Nevertheless, the re-estimation is so rough that it can not distinguish those different unseen n-grams. For example, the word pair "nong2 ye4 (agricultural)" and "jing1 ji4 (economy)" does not co-occur in a Chinese training corpus, neither does another different word pair "nong2 ye4 (agricultural)" and "dang3 (party)", but the former word pair is more reasonable and therefore more possible to appear in new corpus.

Katz [3] develops a modified version of this estimation by combining higher-order models and lower-order models when re-estimating the probability of unseen bi-grams, which is actually a kind of back-off model. According to Equation (2), all the zero-frequency bi-grams are approximated from the relative frequency of those occurring-once bi-grams, each of which is called a singleton. The Katz Smoothing supposes that the occurring probability of the zero frequency bi-gram is also associated with the focused word in the word pair. This idea can be exactly expressed as follows,

$$P_{KS}(v|u)=\begin{cases} d_r(u) f(v|u) & for \ \ r>0 \\ \alpha(u) P_{KS}(v) & for \ \ r=0 \end{cases}, \tag{3}$$

where

$$f(v|u)=\frac{C(u,v)}{C(u)}=\frac{r}{c(u)}, \tag{4}$$

$$d_r=\begin{cases} 1 & for \ \ r>k \\ \dfrac{\dfrac{r^*}{r}-\dfrac{(k+1)n_{k+1}}{n_1}}{1-\dfrac{(k+1)n_{k+1}}{n_1}} & for \ \ k\geq r>0 \end{cases}, \tag{5}$$

and

$$\alpha(u)=\frac{1-\sum_{v:c(u,v)>0} P_s(v|u)}{1-\sum_{v:c(u,v)>0} P_s(v)}. \tag{6}$$

In Equation (4), $c(e)$ is the occurring count of the event e.

Similar to the Katz discounting Equations (3)-(6) for the Bi-gram model, higher order unseen n-grams can also be re-estimated by the (n-1)-Grams recursively. As the end of the recursion, the KS uni-gram re-estimation is taken to be equivalent to the GT re-estimation of this uni-gram.

Compared with the standard Good-Turing re-estimation, Katz Smoothing will assign two different re-estimation values to the word pairs "agricultural economy" and "agricultural party" because of the different occurring frequencies of the words "economy" and "party". Intuitively, this method can result in a better performance than the GT method can.

However, Katz Smoothing has its own shortages. Since the discounting of n-grams with zero and non-zero frequencies is based on different order n-gram information, the results will be contrary to our expectation in a few cases. Considering the two word-pairs (u,v) and (u,w), there may be some unexpected result if the language model is being trained using the corpus where the first word-pair appears only a few times (no more than the predefined threshold k) while the latter one is completely unseen. For example, if the following results are observed for u, v and w:

$$C(u)=5, C(uv)=0, C(uw)=1, P_{KS}(v)=0.2, P_{KS}(w)=0.3$$
$$d_1(u)=0.5, \alpha(u)=0.6.$$

It is thought that the probability of the first pair should be larger than latter one in the KS method since $C(uv)<C(uw)$ and $C(v)>C(w)$. However, according to Equation (3), we could get the smoothed probabilities

$$P_{KS}(v|u)=\alpha(u)\cdot P_{KS}(v)=0.6*0.2=0.12,$$

$$P_{KS}(w|u)=d_1(u)\cdot\frac{C(uw)}{c(u)}=0.4*\frac{1}{5}=0.08.$$

Unfortunately, the result are contrary to our expectation. In other words, the probability is not so smooth as we expect. After browsing our training corpus, we find that this phenomenon is very common, especially in the word-pairs with low counts. One possible solution is to consider the low-order n-gram information in all the cases.

3. PROPOSED APPROACHES

3.1 Enhancing Katz Smoothing by Integrating Deleted Interpolation

The Katz Smoothing can also be expressed in terms of the interpolated model

$$P_{KS}(v|u)=\lambda(u,v)P_{GT}(v|u)+(1-\lambda(u,v))P_{KS}(v). \tag{7}$$

In fact, in order to satisfy the Equation (7) the weight l in the KS method must be chosen such that

$$\lambda(u,v)=\begin{cases} 1 & for \ r>0 \\ \dfrac{\alpha(u)-1}{\beta(u,v)-1} & for \ r=0 \end{cases}, \tag{8}$$

where

$$\beta(u,v)=\frac{1-\sum_{v:c(u,v)>0} P_s(v|u)}{n_0\cdot P_s(v)}. \tag{9}$$

Obviously, the key difference between the back-off and the interpolated models is that the interpolated model considers the information from lower-order n-gram distributions while the back-off model do not when re-estimating the probabilities of n-grams with non-zero counts. This is why the two models perform differently in different corpora. Based on the interpolated form of back-off model, lower-order n-gram distributions can be easily integrated into the Katz Smoothing by redefining the weight function. The modified equation can be rewritten in a uniform formula as

$$\lambda(u,v)=\begin{cases}\mu_r(u,v) & for\ r>0\\ \dfrac{\alpha(u)-1}{\beta(u,v)-1} & for\ r=0\end{cases}. \quad (10)$$

In order that the probabilities of all unseen bi-grams are not altered after the Katz Smoothing, the value of weight function $\mu_r(u,v)$ must be selected to make the accumulated probability of the occurred word pair occurring r times unchanged. That is to say

$$\begin{aligned}&\sum_{v:c(u,v)=r}\left(\mu_r(u,v)\cdot P_T(v|u)+(1-\mu_r(u,v))\cdot P_s(v)\right)\\ &=\sum_{v:c(u,v)=r}P_T(v|u) \qquad \forall r>0\end{aligned}. \quad (11)$$

Generally, we can define the weight function in Equation (10) as a linear interpolated formula of $c(v)$ and $r(=c(u,v))$ as

$$\mu_r(u,v)=\lambda_1(r)+\lambda_2(r)\cdot C(v). \quad (12)$$

For a given frequency r, by substituting Equation (12) for the weight function in Equation (11), we get

$$\begin{aligned}&\lambda_1(r)\cdot\sum_{u,v:c(u,v)=r}\left(P_T(v|u)-P_s(v)\right)+\lambda_2(r)\cdot\sum_{u,v:c(u,v)=r}\left(C(v)\cdot\left(P_T(v|u)-P_s(v)\right)\right)\\ &=\sum_{u,v:c(u,v)=r}\left(P_T(v|u)-P_s(v)\right) \qquad \forall r>0\end{aligned} \quad (13)$$

Therefore we have $2\times R$ pairs of coefficients (i.e., λ_1 and λ_2) and R constrains, where R is the range of r. Then the coefficients can be estimated as the *held-out* algorithm shown in Figure 1:

1. Prepare a kept *corpus* and a *held-out* corpus

2. Train a KS based back-off language model on the *kept* corpus; For each word pair (u,v), store the probability $P_{GT}(v|u)$ and $P_{KS}(v)$ and the occurring time $c(u,v)$ in the *kept* corpus.

3. For each word pair (u,v), compute $N(u,v)$, the number of times that (u,v) takes place in the *held-out* data set;

4. For the occurring time r from *1* to R, try to maximize

$$\sum_{u,v:c(u,v)=r}N(u,v)\cdot\log\left(\mu_r(u,v)P_T(v|u)+(1-\mu_r(u,v))P_s(v)\right)$$

with the limitation of (12) and (13) by the method of

Figure 1. The *held-out* algorithm of parameter estimation

The *held-out* algorithm is often used in the Deleted-Interpolation algorithm [4], therefore the modified model is referred to as the Enhanced Katz Smoothing With Deleted Interpolation (EKSWDI).

3.2 Simplified Versions

EKSWDI can be implemented easily in language decoding. However, we should calculate all the coefficients by many times' executions (to be precise, R times)) of the forth step of the *held-out* algorithm given in Figure 1. Furthermore, the storage perplexity for all the coefficients is a very high. In order to make this model more practical in real-time systems, two simplified versions are presented.

Above all, let us consider the assumptions of these two versions. Firstly, the Katz Smoothing assumes that the probability of frequent bi-grams estimated under the MLE criterion is adequately believable, which leads to the weights in these cases being assigned as constant one with no reduction in performance. Secondly, as we mentioned in Section 2, the lower-order n-gram information is more urgently required to be considered in case r is smaller. According to these two hypotheses, the weight function $\mu_r(u,v)$ can be defined as a normalized relative ratio of the bi-gram probability to the uni-gram probability as in Equation (14).

$$\mu_r(u,v)=\begin{cases}1 & for\ r>k\\ \dfrac{P_T(v|u)}{P_T(v|u)+P_s(v)} & for\ k\ge r>0\end{cases}. \quad (14)$$

This implementation, which is called EKSWRI, can be explained as the relative confidence measure of the bi-gram probability compared with the uni-gram probability. By this formula, the storage perplexity problem is easily overcome because the weights can be obtained dynamically in run-time.

Furthermore, since Equation (14) is still a little bit complicated in the implementation of the language models most commonly used in current speech recognition systems, a simple linear function of the count r as Equation (15) is considered to lower down the time consumption.

$$\mu_r(u,v)=\begin{cases}1 & r>k\\ \dfrac{r}{k+1} & r\le k\end{cases} \quad (15)$$

According to Equation (15), the recursive expression of modified Katz Smoothing can be generalized as follows:

$$P_s(v|u)=\begin{cases}f(v|u) & for\ \ r>k\\ \dfrac{r\left(d_r(u)f(v|u)-P_s(v)\right)}{k+1}+P_s(v) & for\ \ k\ge r>0\\ \alpha(u)P_s(v) & for\ \ r=0\end{cases} \quad (16)$$

However, these two simplifications do not always satisfy the constraint implied in Equation (11). We should re-estimate the value of $\alpha(u)$ and normalize the distributions to make the probability sum-up be one

4. EXPERIMENTS

The language model used in the following experiments is the tri-gram built on a huge corpus that contains about 200 million words. The corpus covers the 4-year's text data of "*People's Daily*" (from 1993 to 1994 and from 1996 to 1997) and a few sections from other Chinese newspaper. The training data are all written texts in news style and in formal language. The vocabulary set consists of 50624 Chinese words, which lengths vary from one to four.

4.1 Perplexity Comparison for Different Models

The perplexities of different models are calculated on a small corpus that contains about 100,000 words. This corpus is also news style and taken from "*People's Daily*" of Year 1999. The results are shown in Table I. In the table, KS means the standard Katz Smoothing back-off model and is used as the baseline. EKSWDI means the model using the algorithm and the weight function discussed in Section 3.1. From the results, we can see that this model achieves a better performance than the KS does with regards to the perplexity. It is because of the improvement of the model's depicting ability achieved by lifting the probabilities of those less-frequent word pairs. EKSWRI means the model using simplified weight function according to the ratio of probabilities from different order n-gram expressed in Equation (14) while EKSWLI means the model using a simplified linear interpolated formula introduced in Equation (15). It is noticed that though the perplexity of MKSWLI is slightly larger than that of MKSWRI or MKSWDI in both bi-gram and tri-gram based experiments, it is still a good choice to make the computation more efficient.

Table I: Perplexity Measure on Different Models

Model	Uni-gram	Bi-gram	Tri-gram
KS	1817	442	274
MKSWDI	-	397	244
MKSWRI	-	402	250
MKSWLI	-	429	267

4.2 Performance Comparison in Chinese Pinyin-to-Character Conversion System

In order to test the performance of the language model and the word decoding procedure without any influence of the recognition errors introduced from the acoustic stage, we develop a system, namely *EasyConv*, to convert Chinese pinyin strings (sentences in toneless pronunciation) into character strings (known as sentences in text). This system takes the correct Chinese pinyin strings of testing sentences as the inputs and then tries to seek the best path in the word graph by the Syllable-Synchronous Network Search (SSNS) algorithm that is based on a kind of modified Viterbi Beam search strategy [5]. And finally, it outputs the putative character strings corresponding to the input pinyin strings. Due to the purpose and architecture of this system, there are only substitution-errors and there won't be any insertion-errors or deletion-errors. The test data is a corpus taken from recent Chinese papers including 1,560 sentences that do not appear in the training corpus. By using the new model, the error rate can be reduced by over 30%. The results are listed in Table II.

Table II: Character error rate (CER) comparison among models in a Chinese Pinyin-to-Character Conversion system

Model	KS	MKSWDI	MKSWRI	MKSWLI
CER	6.5%	4.5%	4.7%	5.0%

5. SUMMARY

The Enhanced Katz Smoothing With Deleted Interpolation is a back-off model integrated with the interpolation of the low-order n-gram information. It assumes that not only the probabilities of unseen n-grams but also those of seen n-grams should be re-estimated according to the low-order n-gram probabilities. This amendment has the following merits:

- **Stronger smoothing ability.**
  It is reasonable to assume that the re-estimation of an n-gram to be smoothed depend on not only the occurring counts of itself and its historical (n-1)-gram but also the frequencies of all its sub-sequences. The GT re-estimation doesn't take account of this hypothesis while the KS does. In the KS algorithm, all the corresponding low-order n-grams are used to smooth the probabilities of unseen n-grams only. The EKSWDI inherits the KS's idea and applies it to all the n-grams no matter whether they appear in the training data or not, which results in stronger smoothing ability.

- **More uniform expression and more flexible expandability**
  The expression of EKSWDI is a uniform one. The Katz smoothing method can be regarded as one of its special cases. The selection of the weight function is entirely free under the constraint of normalized characteristic of the probability sum-up. Therefore, we can choose different functions according to different requirements.

The EKSWDI can be considered as the extension of Katz Smoothing and it is competent for better performance in smoothing the language model probability estimations.

6. REFERENCES

1. F. Jelinek and R. L. Mercer, "Interpolated estimation of Markov source parameters from sparse data", *Pattern Recognition in Practice*, E. S. Gelsema and L. N. Kanal, Eds., 1986, Amsterdam: North-Holland.

2. F. Jelinek and R. L. Mercer, "Probability distribution estimation from sparse data", *IBM Technical Disclosure Bulletin*, 28, 2591-2594.

3. S. Katz, “Estimation of probabilities from sparse data for the language model component of a speech recognizer”, *IEEE Transaction on Acoustic, Speech and Signal Processing*, vol 35, no.3, pp.400-401, March 1987.

4. F. Jelinek, “Statistical methods for speech Recognition”, The MIT Press, Cambridge, Massachusetts, 1997.

5. F. Zheng, “A syllable-synchronous network search algorithm for word decoding in Chinese speech recognition”, *IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP)*, pp. II-601~604, March 15~19, 1999, Phoenix: USA.