Analyzing the Statistical Behavior of Smoothing Method

Feng-Long Huang[1], Ming-Shing Yu[2]

[1]Department of Computer Science and Information Engineering,
National United University,
MiaoLi 360, Taiwan. flhuang@nuu.edu.tw

[2] Department of Information Science, National Chung-Hsing University,
Taichung 40227, Taiwan. msyu@nchu.edu.tw,

ABSTRACT

In the paper, we address the issues for *Good-Turing* smoothing method. Five properties are proposed and employed to analyze the statistical behaviors of *Good-Turing* smoothing method. Because of the violation of the *Good-Turing* smoothing, the related problems should be resolved to keep the method can be used normally.

The problem of zero n_c can be resolved by setting *cut-off k*. The number of n_c and recount c^* of English words and Mandarin character bigrams model are also listed. Three models, *character unigram, bigram and word unigram* model are generated for zero number of n_c .and *bigram* model is used to compare the entropty *H* based on various *cut-off k*.

I. Introduction

The model for word sequence prediction is the n-gram. *n*-gram model uses previous *n*-1 words to predict the next word. In speech recognition, it is always represented as the term language model (LM), [1],[3],[13].

Language models have widely been used in various tasks of natural language processing (NLP), such as speech recognition, machine translation, part-of-speech tagging, spelling correction and word sense disambiguation, etc, [2],[5],[10],[16]. Many research works of language modeling approach, such as [23], [24] and [25], have been used on information retrieval (IR). An event can be regarded as a possible type of *n*-gram in LM, *n*=1, 2, 3. We can calculate the probability for the each occurred event according to its count in training corpora. Because there may be unseen events in testing word sequence, smoothing method should be usually needed to re-estimate prior probability for each event.

A. Language Models

For a source sequence *Str*, the most probable target sequence *W* will be predicted, denoted $w_1,w_2,w_3,\ldots,w_m$ or w_1^m , where *m* is the number of words. Many different word sequences are possible for sequence *Str*.

A language model will be used to decide the correct target word sequence *W*. The prior probability P(*W*), where $W=w_1w_2w\ldots w_m$ is a possible translation of *Str*, can be represented as:

$$P(w_1^m)=P(w_1)P(w_2\mid w_1^1)P(w_3\mid w_1^2)..P(w_m\mid w_1^{m-1})$$

$$=P(w_1)\prod_{i=2}^{m}P(w_i\mid w_1^{i-1}). \quad (1)$$

The prior probability for a given word w_i in Eq. (1) are computed based on a long sequence of preceding words w_1^{m-1} . It is apparent parameter space is so large that a big corpus is needed. One way to resolve this problem is to estimate the approximate probability of a given word by using the $(n\text{-}1)^{\text{th}}$ preceding word sequence. For example, the model with *n*=1 (unigram) can be expressed as:

$$P(w_1^m)=\prod_{i=1}^{m}P(w_i) \quad (2)$$

The probability model with *n*=2 (bigram) can be expressed as:

$$P(w_1^m)=P(w_1)\prod_{i=2}^{m}P(w_i\mid w_{i-1}). \quad (3)$$

As shown in Eq. (3), the probability for each event can be obtained by training bigram model (*n*=2). Therefore the probability of a word bigram *b* will be written as:

$$P(w_i\mid w_{i-1})=\frac{C(w_{i-1}w_i)}{\sum_w C(w_{i-1}w)} \quad (4)$$

where $C(w_i)$ denotes the count of word w_i in training corpus. The probability $P(\cdot)$ of Eq. (4) is the relative frequency and such a method of parameter estimation is called *maximum likelihood estimation* (MLE).

T. Sobh (ed.), Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, 187–192.

B. Unseen Event: Zero count issue

According to definition of the Eqs (2) and (3), the word sequence W can estimated based on MLE. However, such method may lead to the degradation of performance because of unseen events.

For a given word in bigram model, if the so-called the unseen event, which don't occur in training corpora, appear, then $C(w_{i-1}w_i)$ of such event is equal to 0 and Eq. (4) is also equal to 0. It is obvious that the P(W) in Eqs (2) and (3) are 0.

C. Smoothing Motheds for LMs

The unseen events always exist in NLP. Basically, the number of unseen events decreases while size of corpora increases. It is not reasonable to assign 0 to the unseen events. If we should assign certain probability to such unseen events, how the probability is assigned? The probability obtained from MLE should be adjusted and redistributed. Such a process must make the total probability for all known and unseen events to be unity. The schemes used to resolve the problems are called *smoothing methods*; which are usually employed to alleviate the zero-count issue in LMs.

There are many well-known smoothing methods, such as Additive discounting, Good-Turing, Witten-Bell, Katz[1] and so on, [4],[7],[9],[11],[12],[15],[18],[22]. In the paper, we will focus on Good-Turing smoothing and analyze its statistical features for three Mandarin LMs.

II. The Properties for Analyzing Statistical Behaviors of Language Models

In this section, we propose five properties which can be regarded as statistical features of LMs. These properties will be further used to analyze the statistical behaviors of smoothing methods in next section.

1. Property 1

The smoothed probability for any one bigram b_i should falls between 0 and 1 (0,1), as follows:

$$0 < P^*_{i,N} < 1 \quad \text{,for all bigrams } b_i \text{ on any } N \tag{5}$$

where $P^*_{i,N}$ is the smoothed probability for a bigram b_i (or word w_i) on training size N, B is the total number of types of bigrams.

2. Property 2

The summation of smoothed probability P^* for all the bigrams is necessarily equal to 1 on any training size N. Total smoothed probability P is summed as:

$$P^*_{1,N} + P^*_{2,N} + \ldots + P^*_{B,N} = \sum_{b_i \in seen\ bigrams} P^*_{i,N} + \sum_{b_j \in unseen\ bigrams} P^*_{j,N} = 1, \tag{6}$$

where B denotes the total number of bigrams.

3. Property 3

The smoothed probability assigned to the bigrams b with different count should satisfy all the following inequality equations[2]:

$$Q^*_{c,N} < Q^*_{c+1,N}, \quad \text{For } c=0,1,2,\ldots, \tag{7}$$

where $Q^*_{c,N}$ is the smoothed probability for the bigram b_i with c counts on training corpus of size N.

Inequality Eq. (7) describes the concept that smoothed probability for any bigram with same count should be same on any training size N. Furthermore, the probability for bigram b_{c+1} with c+1 counts should be larger than that of bigrams with c counts.

4. Property 4

Comparing to the probability P prior to smoothing process, the smoothed probability P^* for all bigrams will be changed. Property 4 can be expressed as follows:

$$Q^*_{0,N} > Q_{0,N}, \quad \text{for } c = 0 \tag{8}$$

$$Q^*_{c,N} < Q_{c,N}, \quad \text{for } c \geq 1 \tag{9}$$

Property 4 shows $Q^*_{0,N}$ for unseen bigrams will be larger than original

while $Q^*_{c,N}$ will be decreased for all bigrams with more than one count (c>=1). The probability mass P_{mass} discounted from all known bigrams is distributed uniformly to the smoothed probability for unseen bigrams.

5. Property 5

Three notations B, S and U can be expressed as $B=S+U$ for bigram models. When the number of training size is increased, all the smoothed probability Q^* for bigrams with same counts on training size N+1 should be decreased a bit while comparing to the Q^* on training size N. For instance, when an incoming bigram (say b_{N+1}) occurs, the training size is increase by one (now N=N+1). The smoothed probability Q^* on N+1 training set should be less than the probability Q^* on N for $c \geq 0$, except the P^* for the incoming bigram b_{N+1}:

$$Q^*_{c+1,N+1} = \frac{c(\bullet)+1}{N+1}. \tag{10}$$

In other words, in addition to the Q^* of b_{N+1} at training size N+1, all other smoothed probability Q^* at training size N+1 will be decreased than those at training size N. Although both the numerator and denominator of Eq. (10) are increased by 1, due to N>>c, so the inequality equation $Q^*_{c,N} < Q^*_{c+1,N+1}$ will hold. In summary, property 5 can be expressed as:

$$Q^*_{c,N} > Q^*_{c,N+1} \quad \text{for all bigrams with count } c >= 0 \tag{11}$$

$$Q^*_{c,N} < Q^*_{c+1,N+1} \quad \text{for the new bigram } b_{N+1}. \tag{12}$$

[1] In the paper, we employ the *Good-Turing* smoothing method to discount the count c for all events.

[2] The property was first proposed in [17] and we make a little modification.