Smoothing Methods for the Treatment of Digital Texts

*Ruben Dorado**

Fecha de recepción: 28 de febrero 2014
Fecha de aprobación: 05 de marzo 2014
Pp. 43 - 56

Abstract

This article describes the exploration task known as smoothing for statistical language representation. It also reviews some of the state-of-the-art methods that improve the representation of language in a statistical way. Specifically, these reported methods improve statistical models known as N-gram models. This paper also shows a method to measure models in order to compare them.

Key Words

Smoothing methods, treatment, digital texts.

* *Magister en Ciencias de la Computación, Universidad de Tokyo. Pregrado en Ingeniería de Sistemas, Universidad Nacional de Colombia.*

Métodos de suavizado para el Tratamiento de texto digital

Resumen

En este artículo, se describe la tarea de exploración conocida como el suavizado de la representación en lenguaje estadístico. Se hace un estado del arte sobre algunos de los métodos de última generación, que mejoran la representación del lenguaje de una manera estadística. En concreto, estos métodos reportados, mejoran los modelos estadísticos conocidos como modelos de N-gram. Este trabajo también muestra un método para medir los modelos, con el fin de hacer una comparación entre ellos.

Palabras clave

Métodos de suavizado, Tratamiento, Texto digital.

Méthodes de lissage pour le traitement de textes numériques

Résumé

Cet article décrit la tâche dite de lissage de la représentation statistique du langage. Il passe en revue certaines des méthodes les plus récentes améliorant la représentation statistique de la langue. Il s'agit plus précisément des méthodes qui améliorent les modèles statistiques dits « n-gram ». Ce travail rend compte d'une méthode en particulier de mesure et comparaison des modèles entre eux.

Mots clés

Lissage des méthodes, Pour le Traitement de Texte Numérique.

Métodos de alisamento para tratamento de texto digital

Resumo

Este artigo exploratório descreve a tarefa chamada de alisamento para a representação estadística da linguagem. Também revisam-se alguns dos métodos mais recentes que melhoram a representação da linguagem de maneira estadística. Especificamente, os métodos descritos melhoram modelos estadísticos conhecidos como modelos n-gram. Este trabalho também descreve um método para medir modelos com o propósito de compará-los.

Palavras-chave

Métodos de alisamento para tratamento de texto digital.

1. Introduction

Language models are one of the most important elements in the actual current state of language processing of language, specifically in tasks such as machine translation, spelling correction, handwriting recognition and, speech recognition, among many others. A language model represents how the system should treat the relational structure between words, and hence their importance in the field. Probably, the most used language modes are the statistical n-gram based models, which are a straightforward method to represent sequences of words as probabilities. One of the problems these models should overcome are is when an unknown word appears and the model assigns a wrong probability to the whole sentence. Smoothing is a technique used along with n-gram models to better estimate probabilities when unknown words appears for any reason.

This work attempts to measure how the use of a smoothing method can affect the results of language processing for a specific language. In particular, it intends to find the difference of using different smoothing methods for some languages, and in particular for Spanish.

2. Statistical Language Modeling

A language model, and specifically a statistical language model, is formulated as a probability distribution p(s) that assigns a probability to a sequence of n words $s=w_1,\ldots,w_n$. The probability that is assigned to a sequence of words s gives some information about the feasibility of the sequence. For example, a specific language model trained with chat texts can assign p ("hola") = 0,01 since the appearance of the sentence "hola" is one percent out of the whole count of sentences. On the other hand, no plausible sentences such as q = "serpiente trabajó particularmente el lápiz coche" will produce zero values since it is unlikely to see that sequence of words in an actual chat conversation. It is worth to point out that even the sentence q could be grammatically correct and a valid sentence in Spanish, the statistical model will assign a near zero probability.

Most of the statistical language models are based on n-grams. These particular models consider only a window of n-words to approximate the probabilities as explained as follows: given a sentence of n words $s=w_1,\ldots,w_n$, the probability p(s) can be expressed as $p(s)=p(w_1,\ldots,w_n)= p(w_1)\ p(w_2|w_1\ )\ p(w_n|w_1,\ldots,w_{(n-1)})= \prod^{n}_{i=1} p(w_i|w_1,\ldots,w_{(i-1)})$, where $p(w_n |w_1,\ldots,w_{(n-1)})$ represents the probability of a word w_n given its history or the past words in the sentence, and the probability p(s) can be interpreted as the probability of the sentence, s is the product of the probabilities of each word given the previous sequence of words.

N-gram models consider only part of the story taking into account only the previous n words by modelling language as a Markov process of a given order (Markov, 1913). For example, bigram models consider the previous word approximating the whole distribution as:

$$p(s)=\prod^{n}_{i=1)} p(w_i |w_1,\ldots,w_{i-1}) \approx \prod^{n}_{i=1} p(w_i |w_{i-1})$$

To calculate the probability p(s) of an actual sentence or text, the model has to obtain a distribution over words, usually by counting the words and sequences from some texts or training data. In a bigram model, the model has to count how many times a particular bigram w_{i-1}, w_i appears in the text and denoted by $c(w_{i-1},w_i)$. Then, the probability of a particular bigram with a trained model is:

$$p(w_i \mid w_{i-1}) = \frac{c(w_{i-1},w_i)}{c(w_i)}$$

In the special case of using a model of order 2 to calculate the probabilities, where $c(w_i)$ is the total count of appearances of the word w_i. It is possible to generalize the calculation of the probability for a window of l words:

$$p(s)=\prod_{i=1}^{n} p(w_i \mid w_1,\ldots,w_{i-1} \approx \prod_{i=1}^{n} p(w_i \mid w_{i-l}^{i-1})$$

Where w_{i-l}^{i-1} represents the subsequence of words from the positions i-l to i-1, and to calculate $p(w_i \mid w_{i-l}^{i-1})$ from a training data by counting the sequences of words:

$$p(w_i \mid w_{i-l}^{i-1}) = \frac{c(w_{i-l}^{i})}{c(w_{i-l}^{i-1})}$$

3. Smoothing

N-gram models are excellent calculating probabilities with the learned training data, but they do not work when new words come up. More specifically, n-gram models do not take into account grammar and therefore they do not generalize over any structure. For example, if the training data contains the sentence s = “Juan está en camino” the probability p(s) can be calculated as p(Juan)p(está|Juan) p(en|está) p(camino|en). However, if the bigram “Pedro está” does not appear on the training data, then the probability for the phrase “Pedro está en camino” would yield zero since p(está | Pedro)=0. In other words, n-grams models assign wrong values when anto unknown bigrams.

Smoothing techniques are an addition to n-gram models that allow to produce better probabilities with the counts acquired using training data and to avoid zero the probabilities of zero when an unknown n-gram appears. The idea behind smoothing is to adjust the probabilities obtained by training data, making them more uniform by setting low or zero probabilities to be greater and very high probabilities to go downward. The simplest version of smoothing is to add one to each count (Lidstone, 1920; Jeffreys, 1961), yielding to the following way to calculate the probability of a bigram:

$$p(w_i \mid w_{i-1}) = \frac{1+c(w_{i-1},w_i)}{V+c\,(w_i)}$$

Where V is the number of words in the vocabulary. Clearly, if a bigram count is zero then this probability is going to be greater than zero.

4. Smoothing Models for Language Processing

There are several smoothing methods proposed previously. One of the simplest, somehow already introduced in this paper, is called Additive Smoothing or also Laplace Smoothing. The idea behind this method is the same as adding one method but in a generalized mannergeneralize it allowing to specify a quantity α to modify the probability distribution through the following modification:

$$\frac{(\alpha + c(w_{i-1}, w_i)}{(\alpha V + c(w_i)}$$

Where V is the size of the vocabulary and α is generally small $0<\alpha\leq 1$. In the case if α=1 is add one method previously introduced . This method have has several problems that are already identified (Gale & Church, 1994).

The Good-Turing estimate, proposed in Good(Good, 1953) (1953), works by transforming the calculated probability depending on the number of counts. Suppose a word appears r times in the training data, the method works first modifying this count as:

$$r^* \cong (r+1)\frac{n_{r+1}}{n_r}$$

Where n_r represents the frequency of the frequency of number of times n-grams appears in $_r$ times. Finally, it has to be normalized to transform this quantity into a valid probability:

$$p(s) = \frac{r^*}{N}$$

Where $N=\sum_{i=0}^{max(r)} r^* n_r$, and $max(r)$ represents the word that appears most frequently in the training data. It has been reported in other studies (Nádas, 1991) that this method works particularly well when the counts n_r are big.

Other methods include Jelinek-Mercer smoothing (Jelinek & Mercer, 1980) used when training data is particularly small, Katz smoothing (Katz, 1987) which is an extension of Good-Turing estimate, Witten-Bell smoothing (Witten & Bell, 1991) which is similar to Jelinek-Mercer smoothing but this new version takes into account an interpolation of low-order models and high-order models that allows to handle words that have near zero appearances and n-grams that have high counts. Finally, absolute discount (Ney, Esssen, & Kneser, 1994)(Ney, Essen & Kneser, 1994)(Ney, Esssen, & Kneser, 1994) and Kneser-Ney smoothing (Kneser & Ney, 1995) are two similar methods proposed by the same authors that have proven to be very effective. This method works similar to Jelink-Mercer mixing low-order models and high-order models.

5. Evaluation

There are several methods to measure how well a language model performs. Among these, it is common to use perplexity and cross-entropy based measures. All the models presented models are statistical models that assigns a probability p(s) to a sentence s. Then, it is possible to calculate the probability of a set of sentences $S= \{s_1,s_2,\ldots,s_n\}$ as:

$$p(S)=\prod_{i=1}^{n} p(s_i)$$

It is possible to calculate the cross-entropy (Bell, Cleary, & Witten, 1990; Cover & Thomas, 1991), a measure that relates prediction and compression using p(S), the probability of a corpus according to a specific probabilistic model. The cross-entropy of a probabilistic model given a specific corpus of text or training data is defined as:

$$H(p(S))= \frac{-1}{N_s} log_2\, p(S)$$

Where N_s is the number of words in the corpus S. This value measures how well the model is represented by the model p(S) by compressing it using $-\log_2 p(S)$ bits. This measure can also be interpreted as the average number of bits needed to encode the text S. The reciprocal measure is called the perplexity, which is also used as a measure of how well the model is performing. The perplexity is calculated by:

$$PP(S)=2^{H(p(S))}$$

In this case, this number should indicate the average probability assigned by the model to each word in the corpus S. It is important to note that lower values when using cross-entropy and perplexity reflex better models as they indicate better models of representation or compression.

6. Conclusion

This work presents several smoothing models used to improve language models in language processing tasks such as digital text processing or natural language processing. This explorative work aims to describe state of the art methods that improve statistical language models. All the reported models in this work have been tested for English language as it is expected. In future works, we expect to test such models for a particular language and in order to answer the question if they are feasible models for other languages such as Spanish. This work also reviewes the methods that are usually used to test a probabilistic model language that can be employed to compare several models.

7. References

Bell, T., Cleary, J., & Witten, I. (1990). *Text Compression.* Englewood Cliffs: Prentice- Hall.

Cover, T., & Thomas, J. (1991). *Elements of Information Theory.* New Jersey: John Wiley & sons, Inc.

Gale, W. A., & Church, K. W. (1994). *What's wrong with adding one? In N. O. Haan, Corpus-Based Research into Language.* Amsterdam: Rodolpi, Amsterdam.

Good, I. J. (1953). *The population frequencies of species and the estimation of population parameters.* Biometrika, 237-264.

Jeffreys, H. (1961). *Theory of Probability.* Oxford University Press.

Jelinek, F., & Mercer, R. L. (1980). *Interpolated estimation of Markov source parameters from sparse data. Proceedings of the Workshop on Pattern Recognition in Practice*. Amsterdam.

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

Kneser, R., & Ney, H. (1995). *Improved backing-off for m-gram language modeling.* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 181-184.

Lidstone, G. J. (1920). *Note on the general case of the bayes-laplace formula for inductive.* Transactions of the Faculty of Actuaries, 182-192.

Markov, A. (1913). *An example of statistical investigation in the text of 'Eugene Onyegin' illustrating coupling of tests in chains.* Proceedings of the Academy of Science of St. Petesburg, 153-162. St. Petesburg.

Nádas, A. (1991). *On Turing's formula for word probabilities.* IEEE Transactions, Acoustics, Speech and Signal Processing, 825-829.

Ney, H., Esssen, U., & Kneser, R. (1994). *On structuring probabilistic dependences in stochastic language modeling.* Computer, Speech and Language, 1-38.

Witten, I. H., & Bell, T. C. (1991). *The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression.* IEEE Transactions on Information Theory, 1085-1094.