INDICATORS FOR TEXT RELEVANCE USING TECHNIQUES FROM INFORMATION THEORY FIELD

Alina Bogan-Marta,
University of Oradea, Romaina, Department of Computer Sciences,
Armatei Romane Nr 5, Oradea, 3700, alinab@uoradea.ro
Nicolae Robu
"Polytechnica" University of Timisoara
Faculty of Automation and Computer Science and Engineering
Piata Victoria Nr.2, Timisoara, 1900, nrobu@aut.utt.ro

Abstract: Recent years have seen an explosive growth of the volume of text data and the needs of finding relevant information directed research in exploring different approach methods. In this paper we are presenting a work* on finding indicators for text relevance using the estimation of statistical language models. For evaluation there are applied measures from information theory field like entropy/perplexity and the experiments are running on a standard data from WSJ (Wall Street Journal) corpus. The steps folloved in experiments are described in their execution order and at the end, the focus is on the interpretation of the results. Far from pretending that our results are the best, it can be easy observed that they are iportant indicators for text content relevance. Even if this study started from speech recognition perspective, its results can be used as well by many other research fields like that of information retrieval, machine translation, optical character recognition, spelling correction, document classification.

Key words: language models, n-grams, entropy, estimation, evaluation, prediction.

1. INTRODUCTION

In our days, technological trends are to improve machines performances as close as possible to those of human beings. One of the most complex ability of humans is comunication. Writting, speaking, using the signs code determined scientists to search for explanations and clues to understand, explore and exploit these intelligent processes.

A way of exploration can be considered statistical language modeling. On this direction the interest is in determining the probability of naturally occurring word sequences in human natural language.

Traditionally, the dominant motivation for language modeling has come from the field of speech recognition but in the past several years there has been significant interest in the use of language models in many other application areas, as we mentioned already.

*The main work was done during a stay at K.U. Leuven PSI/Speech Group, Belgium, 2003.

The need for a stochastic language model starts from the Bayes' decision rule used in speech recognition to find minimum error rate (Bahl et al., 1983). In this acception if sentence recognition is the goal, than we need to find the optimal word sequence $w_1 \dots w_N$ for which the posterior probability $P(w_1 \dots w_N | x_1 \dots x_T)$ attains its maximum, where $x_1 \dots x_T$ is the sequence of acoustic observations. This rule can be rewritten in the form: $\arg\max_{w_1 \dots w_N} \{P(w_1 \dots w_N) \cdot P(x_1 \dots x_T | w_1 \dots w_N)\}$, where $P(x_1 \dots x_T | w_1 \dots w_N)$ is the conditional probability of observing the sequence of acoustic measurements $x_1 \dots x_T$ given the word sequence $w_1 \dots w_N$ and $P(w_1 \dots w_N)$ is the prior probability of producing the word sequence $w_1 \dots w_N$.

The task of the stochastic language model is to provide estimates of these prior probabilities $P(w_1 \dots w_N)$ (Young , 2001).

For large vocabulary speech recognition, the dependence of the conditional probability of observing a word w_n at a position n is assumed to be restricted to its immediate m predecessor words $w_{n-m} \dots w_{n-1}$. The resulting model is that of a Markov chain and is reffered as (m+1)-gram model . We are describing the applied procedure in section 2 of the paper.

The n-gram is a heuristic concept and its linguistic sense has often been questioned. In practice it has turned out to be extremely powerful and till today it stands out as superior to any formal linguistic approach (Van Compernolle, 2002).

There are various kinds of language models that can be used to capture different aspects of regularities of natural language (Wang et al.).

From the need to compare these language models, to see which better match the corpus or how well a given probabilistic grammar matches human grammars, it is necessary the use of some metrics, measures of information.

Such a measure comes from the information theory field and is the entropy (H). It can be used as a metric for how much information there is in a particular grammar, how well a given grammar matches a language, for how predictive is it, about what the next word/letter could be.

Tightly connected with the entropy is the value 2^H, called perplexity. This measure is intuitively thought of as the weighted average number of choices a random variable has to make (Jurafsky, 2000). Crudely speaking, at the word level, it is a measure of the size of the set words from which the next word is chosen, given that we observe the history of spoken words (Cole et al.,1996).

In this paper we are presenting results of investigations over language models using these measurements and their possible interpretation, reducing the level of application from word to letter|character. The intention is to put in the light the potetial of information theory measures in evaluating large amount of data.

2. COMPUTING THE ENTROPY

As described in (Jurafsky 2000) and (Brown et al ., 1992), computing the entropy requires that we establish a random variable X that ranges over whatever we are predicting (set called ϰ), and that has a particular probability function, call it *P(X)*. The entropy of this random variable X is than

$$H(X) = -\sum_{X \in \chi} P(X) \log_2 P(X). \qquad (1)$$

Because the purpose is to estimate the entropy/perplexity over different language models and to evaluate the real entropy is very difficult in the absence of a very large amount of data, it is a known fact that the cross-entropy of a stochastic process, as measured by a model, is an upper bound on the entropy of the process (Brown et al., 1992). This means that we can use some simplified model *M* to help estimate the entropy. The more accurate *M* is, the closer the cross-entropy *H(P,M)* will be to the true entropy (H(P)).

In this context, the way the cross-entropy was estimated is the sum over the all the n-grams, for which was calculated the product of the estimate of the expectation probability distribution and the conditional probability found in the train set.
The cross-entropy estimate used is described by (2).

$$-\sum_{all\ X_1^n} \hat{P}(X_1 X_2 ... X_n) \log P_M\left(X_1 X_2 ... X_n \mid X_1 X_2 ... X_{n-1}\right) \quad (2)$$

Where X_1^n ranges over all n-grams, $\hat{P}(X_1 X_2 ... X_n)$ is the relative frequency estimate from test set and

$$P_M\left(X_n \middle| X_{-1} X_{-2} ... X_{n-1}\right)$$

is a conditional probability estimated from training set. The convention is:

$$P_M\left(X_n \middle| X_{-1} X_{-2} ... X_{n-1}\right) = 0$$

if for n-gram $X_1 X_2 ... X_{n-1} X_n$ from test set $P_M(X_1 X_2 ... X_{n-1} X_n) = 0$ in the train set, since log0=∞. In such situation, the n-gram will not contribute with any value to the final estimation.
Speculating this weakness, without using any smoothing algorithm, we are exploring the results searching for possible indicators regarding the text corpus relevance.

3. EXPERIMENTS

The tool used in experiments offers the possibility to explore how the described measures of text information behave under different circumstances. It means that we have to describe a map of the required text processing tasks respecting the implementation requirements. After that, the steps followed to evaluate the final entropy (which is the cross-entropy) were: text processing, language model generation, entropy estimation.

Text Processing

A text usually contains not only words but punctuation marks or additional marks which we want or don't want to consider during the evaluation procedure, depending on the proposed task. In this context, for our experiments, the corpus content is kept as much as possible unmodified and, in addition, are marked the simple blank spaces. Assuming that each sentence starts on a new line, are marked the beginning and the end of each sentence also.

The text data used is a collection of 536 text files from WSJ corpus and was split randomly in two sets: one of 183 files (5167 lines, 103970 words, 629966 characters) and another of 352 files (10109 lines, 200724 words, 1215060 characters) respectively.
Over each set, the same preliminary text processing was done so that, from the two text sets considered we obtained one of 615920 characters (5167 lines/sentences), used as *test* corpus, and the other of 1190335 characters (10109 lines/sentences) as *train* corpus.

Generating Language Models

Similar to the classical explanations for n-gram models at word level (Manning, 2000), we applied the same methodology at letter level. The task of predicting the next character can be stated as attempting to estimate the probability function *P*:

$$P\left(X_n \middle| X_1 X_2 ... X_{n-1}\right). \qquad (6)$$

In such a stochastic problem it is used a classification on the previous characters, the *history*, to predict the next one. On the basis of having looked at a lot of text, we can predict which character tends to follow others. In this sense, a method of grouping histories that are similar in some way so as to give reasonable predictions regarding to which character can expect to come next is required. One possible way to group them is by making a Markov assumption that only the prior local context - the last few characters - affects the next one. If we construct a model where all the histories that have the same last

(n-1) elements placed in the same equivalence class, then we have an *(n-1)*th order Markov model or an n-gram character model (the last character of the n-gram being given by the one we are predicting).

Using the files resulted from the first procedure ran over both sets of text, it was generated models for 2,3,4,...,9,12,15,18,20 grams.

Remark: The language models are generated in the frame of each sentence, so that they could be analyzed from syntactic and semantic perspective as well. To simplify the understanding of the n-gram values presented ones has keep track of the fact that the difference between the number of n-grams for each model is a multiple of line/sentence numbers from *test* set.

Because it is necessary to know the *vocabulary size* required, we obtained it from the 1-gram model of the test corpus (here is 74). This means that in addition of those 26 letters from English language (including space) we considered all punctuation marks estimated as relevant for context evaluation. Of course that in many situations there are ambiguities like dots in the middle of the sentence or quotation marks met in condensed form, even if they are not representative on large amount of data, because the purpose was to have an evaluation as close as possible to the real values, the decision was to consider them.

The next stage was running the simple cross entropy for each model.

In the Table 1 can be seen the results obtained and the values are graphically represented with the blue line in Figure 1.

n-grams	Nr.of n-grams in TEST set	Nr.of n-grams in TRAIN set	Nr.of events which do not occur in TRAIN set	Entropy	Perplexity
2	615920	1190355	212	3.55936	11.78894
3	610753	1180246	2153	2.91602	7.54766
4	605586	1170138	10397	2.21442	4.64096
5	600419	1160031	31379	1.63220	3.09987
6	595252	1149933	69916	1.19669	2.29214
7	590086	1139854	123185	0.85135	1.80419
8	584921	1129788	184062	0.58090	1.49579
9	579763	1119730	245694	0.38028	1.30160
12	564326	1089635	388032	0.09469	1.06784
15	548928	1059705	454837	0.02116	1.01478
18	533583	1029945	475142	0.00574	1.00399
20	523397	1010234	476943	0.00306	1.00212

Table 1: The Simple Cross Entropy Evaluation

How can be interpreted this entropy evolution?
The Figure 1 shows that the larger the number of consecutive characters known, the easier to predict the next one is. In our case the simple entropy reaches a zero value after about 11 consecutive characters. A problem could be the evaluation: "is this a good estimation or no?" There is not a standard answer as long as it depends on corpus, topic, and language model used. A true evaluation could be given by the recognition system.

As can be seen in the fourth column of the two tables, a number of n-grams are not participating to the final evaluation. In this case, we want to see whether they are influencing the results of investigations, leading us to a perception of the text data involved. That's why we reconsidered the same experimental procedure but instead of using train corpus for logarithmic evaluation it is used the same test set. In this way we'll have no non participating n-grams. Because the interpretation of the results is a ticklish process and the results differ from corpus to corpus and the premises considered are not always the same, the idea was to find a reference inside of the used corpus.

The new results are in the Table 2 and the graphical representation is marked with red line in the Figure 1 also.

n-grams	Entropy	Perplexity
2	3.54644	11.68382
3	2.87977	7.36034
4	2.16972	4.49936
5	1.61828	3.07010
6	1.22700	2.34081
7	0.92361	1.89686
8	0.67934	1.60142
9	0.48557	1.40014
12	0.16516	1.12129
15	0.05409	1.03821
18	0.02068	1.01444
20	0.01175	1.00818

Table 2: The Entropy Evaluation using overlapping text data

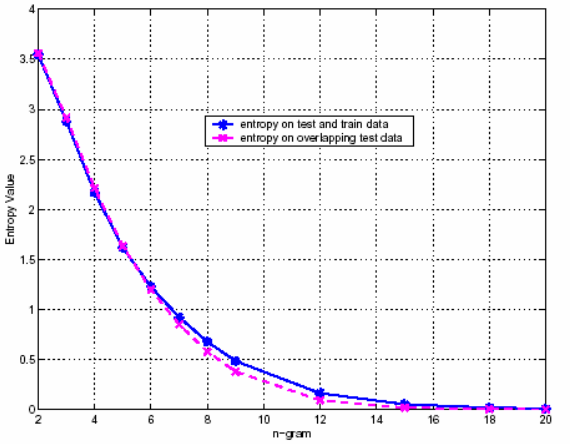

Figure 1 : Entropy evolution generated by overlapping data sets

As it can be seen, comparing the values from the tables and looking at the figure, there is a surprising similar trajectory. The two proposed interpretations we are assuming are: both test and train sets have similar n-gram distribution and the other one is that the non participating n-grams do not influence significantly the final results.

Between the two representations, the main difference is for 7-grams till about 16-grams. If we are considering the Nada's estimation in 1984 (Jurafsky, 2000) regarding the average length of English written words (including space) as 5.5 letters, we can suppose that for sequences greater than 16 and more characters from the vocabulary considered, test and train sets are

sharing many regular expressions. In other words, they belong to the same subject category, or giving little sententious connotations, the authors have the same writing style. In these experiments it seems that the larger the n value is, the closer the two entropies estimation become. Since the corpus dimension is an important aspect, we don't want to go too far with our presumptions but the certain aspects regarding the evaluation of training, testing data are very important to characterize the unread large quantity of texts. In the last years information retrieval techniques are increasingly using these aspects in building the retrieval systems and documents ranking.
In addition, this kind of indicators plays an important role in finding the best corpus to train the speech recognizer system.

4. CONCLUSIONS AND FUTURE WORK

Evaluation is arguably the most important part of any research project (Given,1996). Without proper methods and some widely accepted measures, it is difficult to benchmark one's progress. Evaluation plays an important role for system developers – to tell them if their system is improving - , for consumers – to identify which system best meets their needs. A measure of accuracy, which is not a direct one, is perplexity but being deduced from the entropy we preferred to use the second one for our interpretations.

To evaluate the performance of a language model it has to be done on the same training data as long as it is influenced by many factors. In other words, entropy/perplexity taken out of context has no meaning (Given,1996). That's a reason to search for indicators and on different approaches.

Concluding our presented work, there are resumed the main aspects.

Different than classical approach we are reducing the level of investigations at letter level which allow a better study on a language. Depending on the purpose of the investigation, the punctuation marks, kept in our corpus, may influence the results interpretation. This detailed study helps especially from speech recognition perspective.

Language models generator is built in such a way that all the resulted models contain important syntactic and semantic information. This comes from the fact that every n-gram generated does not count on any history from the previous sentences. The aspect could be helpful as well from word sense disambiguation perspective.

Another aspect is the trial to obtain reference estimation in cross-entropy evaluation running the experiment on identical test and train data. The interpretations do not pretend to be the best possible but we are confident that this approach reveal important aspects regarding the text used both in testing and training of the evaluation system.

Next intention regarding the presented work is to continue investigations on our approach using different corpus. As it was already mentioned, is very difficult to state clear rules in estimation and evaluation of language models because of the large complexity of language itself. However, we can have good indicators, guide lines which give relevant and valuable interpretation clues.

Regarding the accuracy of the results, it would be interesting to compare them with those obtained if we are splitting the used corpus in 4 parts and each of them to be considered test data set and the rest train data set successively.

5. REFERENCES

Bahl, L.R., Jelinek, F., and Mercer, R.L., 1983,"A Maximum Likelihood Approach to Continuous Speech Recognition", in IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 5, pp.179-190.

Brown Peter F., Pietra Stephen A. Della, Pietra Vincent J. Della, Mercer Robert L.L., and Jennifer C.Lai,1992, „An estimation of an upper bound for the entropy of English", in Association for Computational Linguistics, Yorktown Heights, NY 10598, P.O. Box 704.

Cole Ronald A. et all, 1996, „Survey of the state of the art in human language technology", Center for spoken Language Understanding, Oregon, SUA, http://cslu.cse.ogi.edu/HLTSurvey

Given S.P.,1996, „Development of an n-gram based language model for continous speech recognition", The Language Modeling Group, Department of Electrical and Computer Engineering, Spring. http://www.isip.msstate.edu/publications/courses/ece_8463/projects/1996_spring/conference/paper_given.pdf

Jurafsky Daniel and Martin James H., 2000, „Speech and language processing – an introduction to natural language processing, computational linguistics, and speech recognition", pg.191–234, Prentice Hall, Upper Sale River, New Jersey 07458.

Manning Christopher D. and Schütze Hinrich, 2000, „Foundations of statistical natural language processing", Massachusetts Institute of Technology Press Cambridge, Massachusetts London, England, third edition, pg.554 – 556;557 – 588.

Van Compernolle Dirk, „Spoken Language Science and Technology", november 2002, K.U.Leuven-course material.

Young Steve and Gerrit Bloothooft, „Corpus-Based Methods in Language and Speech Processing", Kluwer Academic Publishers, 2001, vol.2, pg.174-179.

Wang Shaojun, Schuurmans Dale, Peng Fuchun, Zhao Yunxin, „Semantic N-gram Language Modeling With The Latent Maximum Entropy Principle". http://citeseer.nj.nec.com/575237.html