Language modeling for sentence retrieval: A comparison between Multiple-Bernoulli models and Multinomial models

David E. Losada
Intelligent Systems Group
Department of Electronics & Computer Science
University of Santiago de Compostela, Spain
dlosada@dec.usc.es

December 15, 2005

Abstract

In this work we focus on a sentence retrieval task to present a comparison between Language Modeling based on a multi-variate Bernoulli distribution and Language Modeling based on the popular multinomial models. Nowadays, a view on text generation as a multiple Bernoulli process is not predominant in Language Modeling for Information Retrieval but we show how the characteristics of the task are appropriate for a statistical Language Modeling based on a multi-variate Bernoulli distribution.

1 Introduction

Although the seminal proposal to introduce Language Modeling in Information Retrieval was based on a multiple-Bernoulli distribution [9], the predominant modeling assumption is now centred on multinomial models. Scoring is simpler in multinomial models and, basically, there is no much evidence giving good reasons to choose multiple-Bernoulli over multinomial in general.

Nevertheless, the characteristics of the task of sentence retrieval suggest that this problem could be addressed by a multiple-Bernoulli approach. The granularity of the task and its particular characteristics seem adequate for a view on text generation as a multiple Bernoulli process.

In this work, we present a comparison between multiple-Bernoulli models and multinomial models in the context of a sentence retrieval task. We conducted a number of experiments showing that a multi-variate Bernoulli model can really outperform popular multinomial models for retrieving relevant sentences.

The rest of the paper is organized as follows. Section 2 sketches the basic foundations of Language Modeling, making special emphasis on bayesian learning with two different likelihood approaches: a multinomial distribution and a multiple Bernoulli distribution. Section 3 reports on the characteristics of the sentence retrieval task and our hypothesis about the adequacy of a statistical modeling with a multiple Bernoulli distribution. Section 4 presents the evaluation conducted in the context of TREC and the paper ends with some conclusions.

2 Language Modeling

Since the pioneering proposal which introduced language modeling for information retrieval [9], different approaches applying statistical language models for supporting the retrieval process have been proposed. The book by Croft and Lafferty [2] presents a good summary of the relevant research undertaken in this area.

One of the most popular methods, the query likelihood approach, is based on 1) estimating an statistical language model for every document and 2) computing the probability of generating the query according to each

document model. Step 1 is a classical statistical learning problem in which we have to estimate a probability distribution (θ_D) from the document's text D, which is treated as a language sample. This kind of problems has been recurrently studied in the context of Bayesian statistics. Indeed, Bayesian learning is a magnificient framework because it does away with the need to explicitly smooth parameters [13].

Given the document D, we can compute the probability of different probability distributions applying the Bayes' rule:

$$P(\theta_D|D) = \frac{P(D|\theta_D)P(\theta_D)}{P(D)} \tag{1}$$

$P(\theta_D)$ encodes our prior belief about the adequacy of the distribution θ_D, $P(D|\theta_D)$ is called the likelihood of the data D under distribution θ_D and $P(\theta_D|D)$ is the posterior distribution. $P(D)$ is the probability of generating the document and it is independent on θ_D.

In order to get the final estimated document model several methods have been used in the literature. A very common approach is to select the single most probable distribution $\widehat{\theta_D}$ that maximizes $P(\theta_D|D)$ (i.e. the mode of the posterior distribution). This is called *maximum a posteriori* (MAP) distribution[1]. Moreover, if we assume that all distributions are equally probable (i.e. we choose a uniform distribution for $P(\theta_D)$) then we get the maximum likelihood estimator, whose problems for IR have been extensively reported in the literature. A different method results from the selection of the expectation of the posterior distribution ($E(\theta_D|D)$) as the estimated document model[2].

The two approaches summarized in the previous paragraph apply different methods for obtaining a single point estimate which is subsequently used for computing the query likelihood. A different approach results from the application of the predictive distribution [13]. This method takes more uncertainty into account because the probability of the query is averaged out under the model over all possible parameters, weighted by the posterior probability. With appropriate choice of the prior distribution this integral can be computed analytically leading to a retrieval formula which can be computed efficiently.

In general, depending on the characterictics of the document in question (e.g. depending on the document length), the posterior distribution $P(\theta_D|D)$ can be narrow or broad and, hence, a single-point estimate might be more or less appropriate. Anyway, an exhaustive comparison between these estimation methods is out of the scope of this work and, in this research, we have opted to work with an MAP approach. Indeed, in the multinomial model, an MAP method with Dirichlet priors leads to the popular Dirichlet smoothing, which is one of the most effective smoothing approaches for IR [14].

2.1 Multinomial distribution

A key modeling decision is the specific form of the likelihood $P(D|\theta_D)$. If we model text as finite sequences of words and we assume that words are generated independently then we get the popular unigram language model, which is a straightforward example of a multinomial distribution:

$$P(w_{s1}, w_{s2}, \ldots, w_{sn}|\theta) = \Pi_{i=1}^{n} P(w_{si}|\theta) \tag{2}$$

This models a random experiment in which the event space is formed by all the possible sequences of words in the vocabulary. The parameters of this multinomial distribution are the probabilities of each different term across the vocabulary: $\{\theta_i\}_{i=1}^{|V|}$, with $\theta_i = P(w_i|\theta)$.

Given the likelihood $P(D|\theta_D)$, different choices of the prior distribution $P(\theta_D)$ may be selected. Nevertheless, for certain choices of the prior, the posterior distribution $P(\theta_D|D)$ has the same algebraic form as the prior.

[1] We can just drop the factor $P(D)$ and take the distribution that maximizes $P(D|\theta_D)P(\theta_D)$.

[2] This estimate ensures that the expected loss computed from the least quadratic error function is minimum [3]. Roughly speaking, this criterium ensures that the average square difference between the estimate and the actual probability distribution is minimum. Of course, in the context IR, the actual probability distribution generating the texts will never be available. Indeed, as argued by Ponte and Croft in [9], the text generation (e.g. query generation) is not a random process but it is treated as such as a means of achieving effective retrieval. Hence, the expectation over the posterior distribution is a conservative way because the *average error* w.r.t an hypothetical ideal distribution is minimum.

Such as choice is called *conjugate prior* and is an algebraic convenience because it simplifies enormously the bayesian analysis. For instance, when $P(D|\theta_D)$ parameterizes a multinomial and $P(\theta_D)$ is Dirichlet (with parameters α_i), which is the conjugate prior for the multinomial distribution, the posterior distribution $P(\theta_D|D)$ is also a Dirichlet distribution:

$$P(\theta_D|D) \propto \frac{\Gamma(|D| + \sum_{i=1}^{|V|} \alpha_i)}{\Pi_{i=1}^{|V|}\Gamma(tf_{i,D} + \alpha_i)} \Pi_{i=1}^{|V|}(\theta_i)^{tf_{i,D}+\alpha_i-1} \tag{3}$$

where Γ is the Gamma function, $|D|$ is the total number of word occurrences in the document, $tf_{i,D}$ is the word count of term i in the document D. The values α_i are hyper-parameters, which can be interpreted as additional data or pseudo-counts associated to each term w_i. The Dirichlet distribution computes a probability associated to each choice of the multinomial parameters, $\{\theta_i\}_{i=1}^{|V|}$, taking into account a set of hyper-parameters α_i, which encode *prior observation counts* for the terms. The posterior distribution is also Dirichlet with parameters $\alpha_i + tf_{i,D}$.

Applying now the MAP approach on the previous equation:

$$\widehat{\theta_D} = \arg\max_{\theta_D} \frac{\Gamma(|D| + \sum_{i=1}^{|V|} \alpha_i)}{\Pi_{i=1}^{|V|}\Gamma(tf_{i,D} + \alpha_i)} \Pi_{i=1}^{|V|}(\theta_i)^{tf_{i,D}+\alpha_i-1} \tag{4}$$

It can be proved that the solution to this equation results in the following estimates for the multinomial parameters:

$$\widehat{\theta_i} = P(w_i|\widehat{\theta_D}) = \frac{tf_{i,D} + \alpha_i - 1}{|D| + \sum_{i=1}^{|V|} \alpha_i - |V|} \tag{5}$$

When the hyper-parameters α_i are set to the same value then we are giving equal preference a priori to all terms in the vocabulary. In particular, setting $\alpha_i = 1$ results in the maximum likelihood estimator and $\alpha_i = 2$ results in Laplace smoothing. On the other hand, setting $\alpha_i = \mu P(w_i|C) + 1$ gives the Dirichlet smoothing[3]. That is, we obtain prior counts for the words across the vocabulary depending on its distribution on some fallback model (e.g. collection model).

Once the $\widehat{\theta_i}$ parameters are set, the retrieval scores are simply obtained by computing the probability that the query text is produced from the estimated document model (query likelihood):

$$P(w_{qt1}, w_{qt2}, \ldots, w_{qtn}|\widehat{\theta}) = \Pi_{i=1}^{n} P(w_{qti}|\widehat{\theta}) \tag{6}$$

Note that document and query are assumed to be generated by the same distribution. This is not the case in other LM approaches such as the seminal approach by Ponte and Croft [9] or the two-stage smoothing proposed by Zhai and Lafferty [15].

2.2 Multiple-Bernoulli distribution

A different approach results from the assumption that texts are bags of words. The number of times a word occurs in a text is not captured. The text generation process can be viewed as a random process in which |V| independent Bernoulli trials are run. In this case, the space of events is very different from the muntinomial case. It is composed of binary vectors whose size is |V|[4].

The form of the likelihood is:

$$\begin{aligned} P(D|\theta) &= \Pi_{w_i \in D} P(w_i|\theta) \Pi_{w_i \notin D}(1 - P(w_i|\theta)) \\ &= \Pi_{i=1}^{|V|}(P(w_i|\theta))^{\delta_{i,D}}(1 - P(w_i|\theta))^{1-\delta_{i,D}} \end{aligned} \tag{7}$$

[3] Observe that α_i are the parameters of the prior probability $P(\theta_D)$, which is Dirichlet, and, thus, its expectation $E(\theta_i)$ is equal to $\frac{\alpha_i}{\sum_i \alpha_i}$. It is intuitive to set the expected value of each θ_i in terms of the probability of the term w_i in the reference collection C.

[4] The Binary Independence Model in the context of classical Probabilistic Models of IR stands on an analagous space of events.

where $\delta_{i,D}$ is 1 if $w_i \in D$ and 0 otherwise. A multiple-Bernoulli formulation for the query generation process was taken by Ponte and Croft in their original application of LM for IR [9]. Nevertheless, the estimation of the document's models, based on a geometric distribution, was somewhat artificial. On the other hand, in the context of Bayesian learning, the interpretation of text generation as a multiple-Bernoulli process leads to a coherent framework in which queries and documents are treated in a uniform way.

Each individual Bernoulli process is governed by its probability of success (i.e. the probability of selecting the term to be included in the generated text), which will be referred to as $\theta_i = P(w_i|\theta)$. The conjugate prior for the multiple-Bernoulli distribution is the multiple-Beta distribution:

$$P(\theta) = \Pi_{i=1}^{|V|} \frac{\Gamma(\alpha_i + \beta_i)}{\Gamma(\alpha_i)\Gamma(\beta_i)} \theta_i^{\alpha_i - 1}(1-\theta_i)^{\beta_i - 1} \tag{8}$$

where α_i and β_i are parameters associated to the Beta distribution. Applying again the MAP distribution:

$$\begin{aligned}
\widehat{\theta_D} &= \arg\max_{\theta_D} P(D|\theta_D)P(\theta_D) \\
&= \arg\max_{\theta_D} \Pi_{i=1}^{|V|} (\theta_i)^{\delta_{i,D}}(1-\theta_i)^{1-\delta_{i,D}} \Pi_{i=1}^{|V|} \frac{\Gamma(\alpha_i + \beta_i)}{\Gamma(\alpha_i)\Gamma(\beta_i)} \theta_i^{\alpha_i - 1}(1-\theta_i)^{\beta_i - 1} \\
&= \arg\max_{\theta_D} \Pi_{i=1}^{|V|} \frac{\Gamma(\alpha_i + \beta_i)}{\Gamma(\alpha_i)\Gamma(\beta_i)} (\theta_i)^{\delta_{i,D} + \alpha_i - 1}(1-\theta_i)^{\beta_i - \delta_{i,D}}
\end{aligned}$$

The solution to this equation results in the following estimates for the Bernoulli parameters:

$$\widehat{\theta_i} = P(w_i|\widehat{\theta_D}) = \frac{\delta_{i,D} + \alpha_i - 1}{\alpha_i + \beta_i - 1} \tag{9}$$

When $\alpha_i = \beta_i = 1$ this results in the maximum likelihood estimator. Smoothed probabilities can be obtained from some fallback model, e.g. $\alpha_i = \mu P(w_i|C) + 1, \beta_i = \frac{1}{P(w_i|C)} + \mu(1 - P(w_i|C)) - 1$[5]. Many other choices of α_i and β_i are possible.

From the estimated $\widehat{\theta_D}$, the query likelihood is computed as:

$$P(Q|\widehat{\theta_D}) = \Pi_{w_i \in Q} P(w_i|\widehat{\theta_D}) \Pi_{w_i \notin Q} (1 - P(w_i|\widehat{\theta_D})) \tag{10}$$

3 Sentence retrieval

There is no much evidence to show that multiple-Bernoulli models can really outperform multinomial models for IR. Although the pioneering proposal was based on a multiple-Bernoulli distribution, multinomial models are now more popular. A key problem in the multi-variate Bernoulli model presented in the last section is that it cannot handle a non-binary notion of term frequency within texts. Indeed, some experiments for the basic retrieval task have been conducted to compare the performance of multiple-Bernoulli models and multinomial models [7] but the results show that multiple-Bernoulli models are not better than multinomial models. Nevertheless, multi-variate Bernoulli models have traditionally performed well for text categorization tasks having a fixed number of attributes [6]. Indeed, Mccallum and Nigam conducted comparative experiments between multinomial and multi-variate Bernoulli for a classification task and they found that multiple-Bernoulli models worked well with small vocabulary sizes but multinomial performs usually better at larger vocabulary sizes [6].

These results inspired us to look at a retrieval task whose characteristics might be adequate for a multiple-Bernoulli approach. In this respect, the task of sentence retrieval could be a good example. Given a query, the process to select relevant sentences from the document base presents some interesting peculiarities. First of all, the lack of a non-binary term frequency component in the multiple-Bernoulli models seems less important

[5] Again, these values come usually after setting the expectation value of θ_i to be equal to the probability of the word w_i in the fallback model, or some variation.

for this task because sentences are short pieces of text. Second, the likelihood used in multiple-Bernoulli models takes into account the non-query terms. For sentence retrieval, this means that the terms in the sentence (especially those ones having $P(w_i|\widehat{\theta_S})$ high[6] which are missing in the query text will produce a *penalty* in the retrieval score. The intuition is that the sentence will probably deviate from the query topic when there are many terms in the sentence which were not mentioned by the query. This could be good for selecting those sentences whose key terms (high probability in the sentence model) are also present in the query. On the other hand, in document retrieval, documents have typically many terms, most of which are non-query terms, and, hence, the benefits from the multiple-Bernoulli likelihood are not evident. To sum up, it seems that the fine granularity of the sentence retrieval task is good for a multiple-Bernoulli formulation.

We will focus on the sentence retrieval task proposed in the context of the TREC novelty track. In this task, the basic input data is a set of TREC topics and a set of relevant documents for each topic. These relevant documents were selected from actual results from an effective retrieval algorithm. If there were 25 or fewer relevant documents for the topic, then all the relevant documents were used. If there were more than 25 documents, the top 25 ranked (and relevant) documents from that run were selected. Participants in the novelty track had to 1) locate those sentences in the relevant documents which are relevant to the topic and 2) filter out those sentences containing redundant material. The purpose of the second step is to provide the user with relevant and *novel* sentences[7]. We focus here our interest in the first stage: sentence retrieval. For each topic, the information available is the text of the topic itself and a rank containing at most 25 relevant documents. Hence, this is a task where the search space (i.e. the set of sentences in these relevant documents) is much smaller than typical search spaces in IR. As argued above, our hypothesis is that multiple-Bernoulli models could be competitive for sentence retrieval in this scenario.

4 Experiments

We tested multinomial and multiple-Bernoulli models for the sentence retrieval problem of the TREC-2002 and TREC-2003 novelty tracks [4, 10]. The characteristics of the TREC-2002 and TREC-2003 data are very different to each other and, hence, we could test the two language modeling approaches under distinct scenarios. In TREC-2002 novelty data, the topics and relevant documents were taken from old TREC tracks (TREC-6, TREC-7 and TREC-8) and a very low percentage of the sentences retrieved in the relevant documents were actually relevant (median around 2%). On the other hand, TREC-2003 participants used the ACQUAINT collection and the topics were constructed specifically for the task. The average percentage of relevant sentences was around 40%. The process to obtain the set of relevant documents (i.e. the input to the novelty participants) and the methods to get the relevance judgments were also changed from TREC-2002 to TREC-2003 [10].

In the experiments, the statistical language models were defined as follows. A language model is defined for each sentence (using eqs. 5 and 9, respectively) and models were smoothed using $\alpha_i = \mu P(w_i|C) + 1$ for the multinomial model (i.e. Dirichlet smoothing) and $\alpha_i = \mu P(w_i|C) + 1, \beta_i = \frac{1}{P(w_i|C)} + \mu(1 - P(w_i|C)) - 1$ for the multiple-Bernoulli model. Different values of the smoothing parameter μ were tested for each model. Scoring is done by query likelihood (eqs. 6 and 10, respectively) and queries are constructed from TREC topics taking all its subfields (title, description and narrative). No stopword processing was done because the effects of stop word removal should be better achieved by exploiting language modeling techniques [14] and, therefore, the comparison between multinomial and multiple-Bernoulli models is not biased by any artificial choice of stopwords. Terms were reduced to its syntactical root with a Porter stemmer.

Two different collection models $P(.|C)$ were used in the experiments. A *poor* fallback model was constructed from the set of relevant documents available for the task (in TREC-2002 novelty track, this is a set of 1080 documents and, in TREC-2003 novelty track, there are 1242 documents available). This simulates an environment in which a large reference collection is not available and the system has to smooth the probabilities from a small set of documents. In the following this collection will be referred to as poor reference collection.

[6] $\widehat{\theta_S}$ is the estimated sentence model which is computed in a similar way as explained in section 2 for document models.

[7] The notion of novelty is captured assuming that the user knows nothing at the time of the initial retrieval and all learning happens in the order of sentence retrieval.

	μ							
	10	100	1k	3k	5k	7k	10k	50k
Multinomial	.058	.172	**.188**	.188	.184	.183	.183	.186
Multiple-Bernoulli	.190	.210	**.214**	.210	.207	.203	.205	.206

Table 1: Multinomial vs multiple-Bernoulli. TREC-2002 - poor reference collection

	μ							
	10	100	1k	3k	5k	7k	10k	50k
Multinomial	.089	.204	.222	.223	**.225**	.223	.225	.222
Multiple-Bernoulli	.211	.221	.222	**.224**	.222	.220	.220	.212

Table 2: Multinomial vs multiple-Bernoulli. TREC-2002 - Large reference collection

On the other hand, we also conducted experiments in which the word statistics were collected from a much larger collection, composed of more than 500k documents from TREC disks #4 and #5. This collection is composed of articles from Finantial Times, Federal Register, Foreign Broadcast Information Service and Los Angeles Times (approx. 2Gb of data) and was used, for instance, in the TREC-8 ad-hoc track [12]. We used the Terrier platform [11] for indexing these documents and we got collection statistics from Terrier's API. The use of these collection models allowed us to compare the relative performance of multinomial and multiple-variate Bernoulli with varying quality of the reference collection. Along this work, this reference collection will be named as large reference collection.

The performance of sentence retrieval algorithms is measured combining sentence set recall and precision through the F measure:

$$F = \frac{2 \times P \times R}{P + R} \tag{11}$$

where P is the fraction of retrieved sentences which are relevant and R is the fraction of the relevant sentences which are retrieved. This is a consistent performance ratio because it is meaningful even when the number of relevant sentences varies widely across topics [4].

In our experiments, we used the top 5% of retrieved sentences for evaluating the TREC-2002 runs and the top 70% of retrieved sentences for TREC-2003. For sentence retrieval applications it will be important to determine an appropriate threshold but threshold tuning was not an objective here. We simply set a threshold taking into account the average percentage of relevant sentences available.

Table 1 depicts performance results for the multinomial and multiple-Bernoulli models in the TREC-2002 sentence retrieval problem (novelty track) with the poor reference collection. Different experiments were run with varying values of the smoothing parameter μ (best results in bold). For all values of the smoothing parameter the multiple-Bernoulli model is better than the multinomial model. On average, multiple-Bernoulli models are 22% better than the corresponding multinomial model. This confirms previous intuitions on the adequacy of the multiple-Bernoulli approach. When the reference model $P(.|C)$ is built from the collection of relevant documents available for the novelty task, the performance of multinomial models is clearly worse than the performance of multiple-Bernoulli models.

In order to check the relative performance for sentence retrieval when a richer reference model is used, we conducted a second pool of tests in which the word statistics were collected from the large reference collection. Results are shown in table 2 (best results in bold).

As expected, when a richer reference model is available the performance of both LM techniques is improved. Moreover, the best performance results of multinomial and multiple-Bernoulli models are now very similar.

Similar experiments were repeated for TREC-2003 data. Results are shown in tables 3 and 4. In TREC-2003 multiple-Bernoulli models are consistently superior to multinomial models. For both reference collections, the multiple-Bernoulli approach gets better performance at all levels of the Dirichlet prior, μ. It is

	μ							
	10	100	1k	3k	5k	7k	10k	50k
Multinomial	.427	.468	.536	.545	.546	.547	.548	**.549**
Multiple-Bernoulli	**.602**	.598	.596	.596	.596	.596	.596	.596

Table 3: Multinomial vs multiple-Bernoulli. TREC-2003 - poor reference collection

	μ							
	10	100	1k	3k	5k	7k	10k	50k
Multinomial	.438	.489	.544	.551	.552	.552	.553	**.554**
Multiple-Bernoulli	**.600**	.598	.597	.597	.597	.597	.597	.597

Table 4: Multinomial vs multiple-Bernoulli. TREC-2003 - large reference collection

interesting to observe that the overall performance of every LM approach is not significantly affected by the reference model applied. It seems that the quality of the reference collection is not an issue for the TREC-2003 data. The high percentage of relevant sentences could make that the performance of the model does not depend strongly on the quality of the fallback model applied. The more relevant sentences available in the collection, the more matching terms between query and sentences and, hence, it seems intuitive to think that the performance will be less dependent on smoothing quality.

The performance of the multinomial model was only comparable to the multi-variate Bernoulli's performance with the TREC-2002 data and a rich reference collection. In all the other cases, the multi-variate Bernoulli was significantly better at retrieving relevant sentences. These results confirm our initial hypothesis about the adequacy of the multiple-Bernoulli model for sentence retrieval. A multiple-Bernoulli formulation seems to be good at isolating the relevant sentences from the non-relevant ones. If a sentence deviates significantly from the query topics (i.e. the sentence model's distribution is concentrated on many non-query terms) then the sentence will be penalized. Although some query terms are actually mentioned by the sentence, if the "focus" of the sentence is located around non-query terms then the retrieval score for the sentence will be low. This effect cannot be obtained through a multinomial model.

We believe that this is a promising result for multiple-Bernoulli models. Retrieval tasks whose granularity is fine could be a good application scenario for LM with multiple-variate Bernoulli formulations.

5 Related work

A similar comparison of multinomial and multiple-Bernoulli models was conducted in [7] but focusing on the classical document retrieval task. They experimented with two multiple-Bernoulli models: Model A, which is the basic multi-variate Bernoulli approach depicted in section 2.2, and Model B, which can handle the number of times that a term appears in a text. In the latter model, the Bernoulli trials are associated to each occurrence of a word in a given location of the text. Although Model B outperforms Model A, its retrieval performance is very similar to the one obtained with a multinomial approach. Hence, they found no good reasons for supporting multi-variate Bernoulli models.

The work by McCallum and Nigam [6] compares the performance of multi-variate Bernoulli and multinomial models for a classification task in different collections and they found evidence on the adequacy of a multiple Bernoulli view when vocabulary sizes were small.

The use of language modeling for the novelty task has been explored in [5]. They applied different LM techniques for both sentence retrieval and novelty detection. For the sentence retrieval task, two different LM methods were tested: a method based on Kullback-Leibler divergence (KLD) between smoothed LMs of a query and a document and a method based on two-stage smoothing. Our interest is different here because we want to explore multi-variate Bernoulli models for sentence retrieval. To the best of our knowledge, no multiple-Bernoulli models were applied for retrieving relevant sentences so far.

Language modeling was also applied for sentence retrieval in the context of a question answering problem [8]. Basically, translation models help to solve the problem of synonymy yielding a better sentence matching process. Translation probabilities between terms are learnt applying methods from machine translation and the subsequent retrieval of sentences shows a significant improvement. The LM approach applied is based on Berger and Lafferty's statistical translation model for IR [1].

6 Conclusions

The experiments reported in this paper are promising for LM based on multi-variate Bernoulli. Sentence retrieval appears as a adequate task in which text generation is viewed as a multiple Bernoulli process. These results encourage us to keep analyzing the role of multi-variate Bernoulli in different IR tasks.

Acknowledgements

I warmly thank Alvaro Barreiro for his useful feedback throughout the development of this research.

References

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proc. SIGIR-99, the 22nd ACM Conference on Research and Development in Information Retrieval*, pages 222–229, Berkeley, USA, August 1999.

[2] W. B. Croft and J. Lafferty. *Language Modeling for Information Retrieval*. Kluwer Academic, 2003.

[3] M. deGroot. *Probability and Statistics*. Addison-Wesley, 1988.

[4] D. Harman. Overview of the trec 2002 novelty track. In *Proc. TREC-2002, the 11th text retrieval conference*, 2002.

[5] L. Larkey, J. Allan, M. Connell, A. Bolivar, and C. Wade. Umass at trec 2002:cross language and novelty tracks. In *Proc. TREC-2002, the 11th text retrieval conference*, 2002.

[6] A. Mccallum and K. Nigam. A comparison of event models for naive bayes text classification. In *Proc. AAAI/ICML-98 Workshop on Learning for Text Categorization*, pages 41–48. AAAI press, 1998.

[7] D. Metzler, V. Lavrenko, and W. B. Croft. Formal multiple-bernoulli models for language modeling. In *Proc. 27th ACM Conference on Research and Development in Information Retrieval, SIGIR'04*, pages 540–541, Sheffield, UK, 2004. ACM press.

[8] V. Murdock and W.B. Croft. Simple translation models for sentence retrieval in factoid question answering. In *Proc. ACM SIGIR Workshop on Question Answering*, Sheffield, UK, 2004.

[9] J. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proc. 21st ACM Conference on Research and Development in Information Retrieval, SIGIR'98*, pages 275–281, Melbourne, Australia, 1998.

[10] I. Soboroff and D. Harman. Overview of the trec 2003 novelty track. In *Proc. TREC-2003, the 12th text retrieval conference*, 2003.

[11] TERRIER information retrieval platform. http://ir.dcs.gla.ac.uk/terrier/.

[12] E. Voorhees and D. Harman. Overview of the eight text retrieval conference. In *Proc. TREC-8, the 8th text retrieval conference*, 1999.

[13] H. Zaragoza, D. Hiemstra, and M. Tipping. Bayesian extension to the language model for ad hoc information retrieval. In *Proc. 26th ACM Conference on Research and Development in Information Retrieval, SIGIR'03*, pages 4–9, Toronto, Canada, 2003.

[14] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to adhoc information retrieval. In *Proc. 24th ACM Conference on Research and Development in Information Retrieval, SIGIR'01*, pages 334–342, New Orleans, USA, 2001.

[15] C. Zhai and J. Lafferty. Two-stage language models for information retrieval. In *Proc. 25th ACM Conference on Research and Development in Information Retrieval, SIGIR'02*, pages 49–56, Tampere, Finland, 2002.