Experiments with N-Gram Prefixes on a Multinomial Language Model versus Lucene's off-the-shelf ranking scheme and Rocchio Query Expansion (TEL@CLEF Monolingual Task)

Jorge Machado[1], Bruno Martins[1], José Borbinha[1]
[1] Departmento de Engenharia Informática, Technical University of Lisbon, Portugal.
{jorge.r.machado, bruno.martins, jose.borbinha}@ist.utl.pt

Abstract. We describe our participation in the TEL@CLEF task of the CLEF 2009 ad-hoc track, where we measured the retrieval performance of LGTE, an index engine for Geo-Temporal collection which is mostly based on Lucene, together with extensions for query expansion and multinomial language modelling. We experiment an N-Gram stemming model to improve our last year experiments which consisted in combinations of query expansion, Lucene's off-the-shelf ranking scheme and the ranking scheme based on multinomial language modeling. The N-Gram stemming model was based in a linear combination of N-Gram, with N between 2 and 5, using weight factors obtained by learning from last year topics and assessments. The Rocchio ranking function was also adapted to implement this N-Gram model. Results show that this stemming technique together with query expansion and multinomial language modeling both result in increased performance.

Keywords: Language Model, Vector Space Model, Lucene, Rocchio QE, Porter Stemming, N-Grams Stemming.

1 Introduction

One task of the ad-hoc track at the 2009 edition of the Cross Language Evaluation Forum (CLEF) addresses the problem of searching and retrieving relevant items from collections of bibliographic records from The European Library (TEL@CLEF). Three target collections were provided, each corresponding to a monolingual retrieval task where we participated: the TEL Catalogue records in English (Copyright British Library), the TEL Catalogue records in French (Copyright Bibliothèque Nationale de France) and finally the TEL Catalogue records in German (Copyright Austrian National Library). The evaluation task aimed at investigating the best approaches for retrieval from library catalogues, where the information is frequently very sparse and often stored in unexpected languages. This paper describes the participation of the Technical University of Lisbon at the TEL@CLEF task. Our experiments aimed at measuring the retrieval performance of the LGTE[1] tool [8] [4], the IR service of DIGMAP[2], using stemming techniques to turn the system language independent and robust with degraded texts resulting from OCR processes. In case of success, the ultimate goal of the project is to become fully integrated into The European Library which aims to index OCR texts in multiple languages.

Like last year in CLEF, we experimented with combinations of query expansion, Lucene's off-the-shelf ranking scheme and the ranking scheme based on multinomial language modeling. But this year we also included an N-Gram model consisting in a linear combination of independent indexes, each containing stemmed tokens of different grams. The technique was proposed by Parapar in [9] and aims to improve retrieval in degraded collections. Our main objective was to have a language independent model which also could be used with bibliographic metadata records, which of course are not degraded. This paper is structured as follow: first we review the related word in ranking schemes used in this experiment and the related work with the Rocchio's algorithm. Second we describe our ranking scheme and the modifications purposed for the Rocchio's algorithm in order to take benefit of our scheme. In third place we describe our experimental story and discuss the obtained results. Finally we present conclusions.

2 Related Work

The underlying IR system used in our submissions is based on Lucene[3], together with a multinomial language modeling extension developed at the University of Amsterdam, a linear combination of scores calculated in independent indexes of words stemmed with N-Grams technique, and finally a query expansion extension developed by Neil Rubens. The following subsections detail these components.

2.1 Lucene's off-the-shelf retrieval model

We started with Lucene's off-the-shelf retrieval model. For a collection D, document d and query q, the ranking score is given by the formula bellow:

$$ranking(q,d) = \sum_{t \in q} \frac{tf_{t,q} \cdot idf_t}{norm_q} \cdot \frac{tf_{t,d} \cdot idf_t}{norm_d} \cdot coord_{q,d} \cdot weight_t \tag{1}$$

where:

$$tf_{t,X} = \sqrt{termFrequency(t,X)}, \quad norm_q = \sqrt{\sum_{t \in q} tf_{t,q} \cdot idf_t^2}, \tag{2}$$

$$idf_t = 1 + \log \frac{|D|}{documentFrequency(t,D)}, \quad norm_d = \sqrt{|d|},$$

$$coord_{q,d} = \frac{|q \cap d|}{|q|}$$

[1] http://code.google.com/p/digmap/wiki/LuceneGeoTemporal

[2] http://www.dgmap.eu

[3] http://lucene.apache.org

Lucene has been extensively used in previous editions of the CLEF, NTCIR and TREC joint evaluation experiments.

2.2 Lucene extension based on multinomial language modeling

We experimented with an extension to Lucene that implements a retrieval scheme based on estimating a language model (LM) for each document, using the formula described by Hiemstra [2]. This extension was developed at the Informatics Institute of the University of Amsterdam[4]. For any given query, it ranks the documents with respect to the likelihood that the document's LM generated the query:

$$ranking(d,q) = P(d \mid q) \propto P(d) \cdot \prod_{t \in q} P(t \mid d) \qquad \textbf{(3)}$$

In the formula, *d* is a document and *t* is a term in query *q*. The probabilities are reduced to rank-equivalent logs of probabilities. To account for data sparseness, the likelihood *P(t|d)* is interpolated using Jelinek-Mercer smoothing:

$$P(d \mid q) = P(d) \cdot \prod_{t \in q} ((1-\lambda) \cdot P(t \mid D) + \lambda \cdot P(t \mid d)) \qquad \textbf{(4)}$$

In the formula, *D* is the collection and λ is a smoothing parameter (in our experiments it was set to the default value of 0.15). The model needs to estimate three probabilities: the prior probability of the document, *P(d)*; the probability *P(t|d)* of observing a term in a document, and the probability *P(t|D)* of observing the term in the collection. Assuming the query terms to be independent, and using a linear interpolation of a document model and a collection model to estimate the probability of a query term, the probabilities can be estimated using maximum likelihood estimates:

$$P(t \mid d) = \frac{termFrequency(t,d)}{|d|} \qquad P(d) = \frac{|d|}{\sum_{d' \in D} |d'|} \qquad \textbf{(5)}$$

$$P(t \mid D) = \frac{documentFrequency(t,D)}{\sum_{t' \in D} documentFrequency(t',D)}$$

This language modeling approach has been used in past experiments within the CLEF, NTCIR and TREC joint evaluation campaigns – see for example [6].

2.3 Rocchio query expansion

The fact that there are frequently occurring spelling variations and synonyms for any query term degrades the performance of standard techniques for ad-hoc retrieval. To overcome this problem, we experimented with the method for pseudo feedback query expansion proposed by Rocchio [3]. The Lucene extension from the LucQE project[5] implements this approach. On test data from the 2004 TREC Robust

[4] http://ilps.science.uva.nl/Resources/

[5] http://lucene-qe.sourceforge.net/

Retrieval Track, LucQE achieved a MAP score of 0.2433 using Rocchio query expansion. Assuming that the top *D* documents returned for an original query q_i are relevant, a better query q_i+1 can be given by the terms resulting from the formula bellow:

$$q_{i+1} = \alpha \cdot q_i + \frac{\beta}{|D|} \cdot \sum_{d_r \in D} termWeight(d_r) \tag{6}$$

In the formula, α and β are tuning parameters. In our experiments, they were set to the default values of, respectively, 1.0 and 0.75. The system was trained through experiments with the 2008 AdHoc topics and relevance judgments. We found an optimal value of 64 expansion terms for English topics and 40 expansion terms for French and German topics. The terms were extracted from the highest ranked documents (i.e. the $|D|$ parameter) from the original query q_i. With the training we obtain optimal values using 7 documents for English and French and 8 documents for the German collection.

2.3 Linear Combination of N-Grams

The stemming technique based on N-Grams is very popular with texts produced from OCR (Optical Character Recognition) processes, because many times they bring errors. This technique consists in tokenizing the words with a sliding window into tokens of size N, with N assuming several sizes. This process is applied both in documents and queries to increase retrieval performance. Recent experiments related in [9] by Parapar demonstrate that using independent N-Grams indexes, for example from 2 to 5 grams, and combining the individual ranks in a linear combination, can improve the results when we find good parameter values to weight each independent score. The final score is illustrated by the formula 6, as introduced in [9].

$$s(d) = \alpha \times s_{term}(d) + \beta \times s_{5gram}(d) + \gamma \times s_{4gram}(d) + \delta \times s_{3gram}(d) + \epsilon \times s_{2gram}(d) \tag{7}$$

In this formula *d* is the document, s_{Nterm} is the score of that term in the index of grams with size N. Parameters α, β, γ, δ and ε are the weights assigned to each independent score.

2 Ranking Scheme

The following subsections detail how we adapted the ranking scheme based on the combination of N-Grams for bibliographic records and also the modifications in Rocchio query expansion algorithm in order to take benefit of our ranking scheme.

2.1 N-Gram ranking scheme

The original N-Grams stemming, which tokenizes the words with a sliding window, does not fit very well in our problem because our records were not obtained from OCR processes (so we don’t have characters errors). On other hand using this technique turns the stemming phase a language independent process, which was our main focus. For that reason, we used a simplistic approach for the N-Grams model which consists in suffixes removal starting in character N+1. We used an “N-length stemming” where N is the size of the indexed prefix (e.g. *stem-5(“retrieval”) = “retri”*). We tokenized our terms in five different ways, each to produce a different index file. We created four indexes, for the cases of 2-grams, 3-grams, 4-grams and 5-grams, and one other with the original terms. As an example, let us consider a document with the word “retrieval”. That document will be indexed as follows: originalTerms: *retrieval*, 5-grams: *retri*, 4-grams: *retr*, 3-grams: *ret*, 2-grams: *re*. Referring to the weight parameters presented in previous section our system was trained through experiments with CLEF 2008 AdHoc topics and relevance judgments to optimize the Mean Average Precision (MAP). Table 1 shows the optimal values found for each index in each collection. We found that bi-grams worsen the results so we set their weight to zero in the three evaluated collections.

Table 1. Weight values found for each index using MAP in 2008 relevance judgments.

Lanuage	***α***	***β***	***γ***	***δ***
English	0.45	0,27	0,25	0,03
French	0.53	0,24	0,22	0,01
German	0,55	0,23	0,21	0,01

2.2 N-Grams and Rocchio query expansion

In order to deal with N-Gram prefix stemming we had the need to adapt the Rocchio formula. Originally, the Rocchio algorithm calculates the ranking for the terms of the top documents with the formula (7) and selected, for the expanded query, the highest ranked terms boosting them in the final query with the obtained rank. Our issue was how to do that considering that we have five indexes instead of one. Three techniques were experimented but only the third one improved the results. The first and second attempt could be consulted in the working notes of this paper published in CLEF. Our best approach consisted in the following steps. For each one of our top ranked documents *D* we proceeded as follows: **First** of all, using each one of the 5 independent indexes ({2,3,4,5}grams plus original terms index), the system scored all the document terms *t* present in those indexes using the follow formula:

$$Score(t,index_t) = TF.IDF(t,index_t)*decay(D)*weight(index_t) : with\ t\ in\ index_t \qquad (\mathbf{8})$$

In this formula *decay(D)* is the decay factor related with document *D* position in the retrieved list. The *weight* is the factor found for that term index (Table 1). **Second**, the scored terms from all the 5 indexes were sorted in one unique list, independently of the source index. **Finally,** we created the expanded query using the original terms of the query, boosted by 1, plus the top ranked terms in the sorted list boosted with the score of the term. This method weaked the tokens from less weighted indexes like 2-Grams and 3-Grams. This resulted that tokens from weaker indexes could only be picked if they were very relevant in their own indexes. Expanded queries were mainly composed by tokens of 4 or 5-grams and original terms, but all queries had tokens from all indexes, even the weakeast ones.

3 The experimental story and obtained results

We aimed to experiment the performance of Porter stemming technique versus the linear combination of N-Grams, with and without query expansion, using two different ranking schemes for text: the Vector Space Model and the Multinomial Language Model. Our objective was to optimize several parameters to maximize the MAP measure using CLEF 2008 AdHoc topics and relevance judgments. For each collection (EN, FR, DE) we optimized the parameters of Rocchio technique and the weights assigned to each independent index of 3, 4, 5 grams tokens and the not stemmed words index (original terms). The optimized values were already presented in the sections Related Work and Ranking Scheme. The optimized values were used to run the 2009 topics.

Before the indexing, the documents (i.e. the bibliographic records) and the topics were passed through the following pre-processing operations. **Field weighting** of the bibliographic records was applied using the scheme proposed by Robertson et. al [5] to weight the different document field according to their importance. The combination used in our experiments was based on repeating the *title* field three times, the *subject* field twice and keeping the other document fields unchanged. We also **normalized** topics and collection reducing all characters to the lowercase unaccented equivalents (i.e. "Ö" reduced to "o" and "É" to "e" etc.). We also removed **stopwords** using lists from the Snowball package[6]. We **stemmed the words** of the documents using, in first experiment, Porter [1] stemming algorithm from the Snowball[6] package, specific to the language, and in the second experiment using tokens of length 3, 4 and 5 plus the original words in five independent indexes. The **topic processing** was fully automatic including two times the title, one time the description and we didn't use the narrative. In the topics, the resultant words were also stemmed using the Porter technique or stemmed to tokens of length 3, 4 and 5 plus the original words. In the second case the queries were split into five parts, each boosted by the optimized values enumerated in Table 1 (section 2.1). Take as an example the topic "Title: *Adhoc;* Description: *information retrieval*" for English collection, the result query is given by:

> "*words:(adhoc adhoc information retrieval)^0.45 g5:(adhoc adhoc infor retri)^0.27 g4:(adho adho info retr)^0.25 g3:(adh adh inf ret)^0.03.*"

In the query the labels *words*, *g5*, *g4 g3* are indexes and ^*x* is a boost factor. The Lucene-LM[4] machine was adapted to calculate independently each part of the query in order to implement the linear combination of N-Grams detailed in previous sections.

2.2 Results

We now present the complete set of experiments for the three languages using the two text models, vector space and language model, and combining Rocchio query expansion with the two stemming approaches. Table 2 shows the obtained results in

[6] http://snowball.tartarus.org/

terms of the mean average precision (MAP), precision at first five results (P@5) and precision at first 10 results (P@10). In the table, VS means Vector Space Model, LM means Language Model and QE means Query Expansion.

Table 2. MAP vs. MAP 2008 optimization, P@5 and P@10 for all the combinations.

				English				French				German			
	Model	Stemming	QE	MAP	MAP **2008**	P@5	P@10	MAP	MAP **2008**	P@5	P@10	MAP	MAP **2008**	P@5	P@10
1	*VS*	*no*	*no*	0.3403	>0.3242	0.6360	0.5200	0.2030	<0.2352	0.4400	0.3380	0.1357	>0.1290	0.3080	0.2340
2	*LM*	*no*	*no*	0.3496	>0.3228	0.6480	0.5260	0.2255	<0.2412	0.4680	0.4020	0.1480	<0.1523	0.3160	0.2680
3	*VS*	*Porter*	*no*	0.3710	<0.3789	0.6320	0.5500	0.2338	<0.2561	0.4360	0.3640	0.2372	>0.2132	0.4920	0.3720
4	*LM*	*Porter*	*no*	0.3829	<0.3914	0.6800	0.5480	0.2647	<0.2781	0.4760	0.3860	0.2473	>0.2326	0.5040	0.3880
5	*VS*	*Grams*	*no*	0.3966	>0.3750	0.6760	0.5620	0.2508	<0.2967	0.4800	0.4000	0.2439	>0.2306	0.4800	0.3680
6	*LM*	*Grams*	*no*	0.3902	>0.3775	0.6800	0.5500	0.2526	<0.2821	0.4960	0.4080	0.2524	>0.2266	0.4880	0.3880
7	*VS*	*no*	*Rocchio*	0.3712	>0.3526	0.6240	0.5400	0.2015	<0.2640	0.4320	0.3420	0.1725	<0.1875	0.3320	0.2740
8	*LM*	*no*	*Rocchio*	0.3778	>0.3695	0.6200	0.5420	0.2213	<0.2759	0.4280	0.3500	0.1921	>0.1913	0.3320	0.3060
9	*VS*	*Porter*	*Rocchio*	0.4012	>0.3980	0.6640	0.5560	0.2186	<0.2517	0.4240	0.3380	0.2810	**>0.2629**	0.5400	0.4100
10	*LM*	*Porter*	*Rocchio*	0.4143	**<0.4306**	0.6960	0.5920	0.2391	<0.2722	0.4240	0.3500	0.2891	>0.2586	0.5160	0.4400
11	*VS*	*Grams*	*Rocchio Grams*	**0.4393**	>0.4088	0.6760	0.5720	**0.2641**	**<0.3261**	0.4760	0.3880	**0.3005**	**>0.2813**	0.5080	0.4240
12	*LM*	*Grams*	*Rocchio Grams*	**0.4240**	**>0.4140**	0.6720	0.5680	**0.2653**	**<0.3021**	0.5120	0.4100	**0.3049**	**>0.2600**	0.5240	0.4160

The weighted model of N-Grams allied with the Rocchio query expansion outperforms almost all the other configurations in all languages. Using Rocchio the model was better than the text models allied with Porter stemming technique, otherwise was very similar. Statisticaly comparing the MAP of baselines (runs 1 and 2) with runs 5 and 6 using grams the t-test's returns in all collections less than 0,0005 except on French run 6 where return 0,019 which is also good. Using runs 7 and 8 (Porter plus Rocchio) as baselines we tested the significance of using N-Grams plus Rocchio (runs 11 and 12) and the t-test show significance in all results returning less than 0,015 except for Language Model in English (run 12) returning 0.2089.

These results gave us a strong hope on how to build language independent retrieval systems. Comparing with the other participants of the AdHoc task this experiment obtained the better MAP in English and the third better in French and German. In the English and German collections the MAP results outperform the optimized MAP for the 2008 topics what is impressive and prove that the model is very strong. The French collection is the only one where the results loose significantly (~5%) for the optimized ones. We need to perform more evaluation to check this result but we also found that the problem is general to all the other participations in AdHoc track.

5 Conclusions

The obtained results support the hypotheses that using Rocchio query expansion together with a weighted model of N-Grams and a ranking scheme can be beneficial to the CLEF ad-hoc task. Applying this technique to bibliographic records using the prefix stemming instead of a sliding window to tokenize words outperform the Porter stemming technique in most scenarios, especially when the linguistic stemmers are not appropriate. Using this technique with different text models, vector space or language model, is independent because terms are scored independently. Unlike last year where our experiments resulted in poor results for both the French and German collections, this year we could obtain very encouraging results. Like last year we realize that multinomial language model performs almost equal to vector space model in most of the situations. On other hand the multinomial language model has the advantage that we could train it very easily just by tuning the language model parameters, which was not our objective in this experiment, so we believe that language model has potential to return even better results than vector space model.

The results obtained in this experiment support the future implementation of this model in the TEL (The European Library) search service with full text. That comprised degraded texts resultant from OCR of digitized works, to be provided by all the TEL partners, consequently in several languages, which fits very well with the purpose of this model.

References

1. Porter, M. F.: An algorithm for suffix stripping: In: Sparck Jones, K. & Willett, P. (eds.), Readings in Information Retrieval., pp. 313 - 316. San Francisco: Morgan Kaufmann. (1980)
2. Hiemstra, D.: Using Language Models for Information Retrieval: Ph.D. Thesis, Centre for Telematics and Information Technology, University of Twente. (2001)
3. Rocchio, J. J.: Relevance Feedback in Information Retrieval: In: The SMART Retrieval System. Experiments in Automatic Document Processing: pp 313 - 323. Prentice Hall. (1971)
4. Machado, J.: Mitra: A Metadata Aware Web Search Engine for Digital Libraries: M.Sc. Thesis, Departamento de Engenharia Informática, Technical University of Lisbon. (2008)
5. Robertson, S., Zaragoza, H., and Taylor, M.: Simple BM25 extension to multiple weighted fields: In Proceedings of the Thirteenth ACM international Conference on information and Knowledge Management (Washington, D.C., USA, November 08 - 13, 2004). CIKM '04. ACM, New York, NY, 42-49. (2004)
6. Ahn, D. D., Azzopardi, L., Balog, K., Fissaha, A. S., Jijkoun, V., Kamps, J., Müller, K., de Rijke, M. and Erik Tjong Kim Sang: The University of Amsterdam at TREC 2005: Working Notes for the 2005 Text Retrieval Conference. (2005)
7. Pedrosa, G., Luzio, J., Manguinhas, H., and Martins, B.: DIGMAP: A service for searching and browsing old maps: In Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries (Pittsburgh PA, PA, USA, June 16 - 20, 2008). JCDL '08. ACM, New York, NY, 431-431. (2008)
8. Machado J, Martins B, Borbinha J., "LGTE: Lucene Extensions for Geo-Temporal Information Retrieval", European Conference on Information Retrieval, at Workshop on Geographic Information on Internet, Toulouse, April 2009.
9. Parapar, Javier; Freire, Ana; Barreiro, Álvaro (2009). "Revisiting N-gram Based Models for Retrieval in Degraded Large Collections", European Conference on Information Retrieval, Toulouse, April 2009