UB at TREC-13: Genomics Track

Miguel E. Ruiz[1], Munirathnam Srikanth[2], and Rohini Srihari[1,2]

[1] State University of New York at Buffalo
School of Informatics,Dept of Library and Information Studies
534 Baldy Hall
Buffalo, NY 14260-1020 USA
meruiz@buffalo.edu,
WWW home page: http://www.informatics.buffalo.edu/faculty/ruiz
[2] Dept. of Computer Science and Engineering
201 Bell Hall Box 60200
Buffalo, NY 14228
srikanth@languagecomputer.com
rohini@cedar.buffalo.edu

Abstract. This paper describes the experiments of the State University of New York at Buffalo in TREC 13. We participated in the Genomics track and submitted official runs to the Adhoc retrieval task. Our approach uses a language model IR system developed in house. We also present unofficial results for the triage sub-task of categorization task.

1 Introduction

For TREC 2004 our group participated in the Genomics track. Our Adhoc retrieval work used the statistical language model system TAPIR (Text Processing and Information Retrieval). This is a toolkit is of software tools that facilitate a number of IR tasks and supports different IR models including language models. TAPIR was developed in house by M Srikanth. TAPIR has been used previously in TREC-12 on the HARD track [3] and we thought that this could be a good opportunity to test it on a large domain specific collection.

2 Statistical Language Models for Domain Specific Collections

Statistical language models have been shown to be very effective for document retrieval [1] [4]. A language model is a probability distribution defined on strings of an alphabet. A language model is associated with a document in the document collection to indicate or capture its unique properties. Given a query, Q, the documents are ranked based on the likelihood of their language model generating the query,$P(Q|M_d)$ [2]. The query-likelihood probability is estimated using smoothed unigram language models.

$$P(Q|M_d) = \prod_i P(q_i|M_d) \tag{1}$$

The query term probability is estimated from document and corpus counts of the query term smoothed using Dirichlet priors. In Bayesian smoothing using Dirichlet priors, the language model is assumed to be multinomial with the conjugate prior for Bayesian analysis as the Dirichlet distribution $\{\mu P_C(w_i)\}$. The Dirichlet prior smoothed term probability is given by

$$P(w|M_D) = \frac{n(w,d) + \mu p_C(w)}{\sum_v n(v,d) + \mu} \tag{2}$$

where μ is the Dirichlet prior parameter, $n(w,d)$ is the count of occurrence of term w in document d. $p_C(w)$ is the corpus probability of term w. A fixed value of $\mu = 1000$ was used in the experiments.

2.1 Results and Analysis

We submitted a single run for the Adhoc retrieval task (UBgtNormJM1). For this run we use a smoothed unigram model. For processing the text we use a simple stemmer that recognizes plurals only and a stop-word list adapted to the biomedical domain. The current configuration of our system does not use retrieval feedback or other advanced retrieval features. The average precision for this run is 0.2043 and an R-precision of 0.2510. Figure 1 shows performance with respect to the average system reported in the official runs. In general, this slightly below average but our model uses a simple retrieval model with no pseudo-relevance feedback mechanism. Our main goal, which was to test the system with a large domain specific collection, was fulfilled with this experiment.

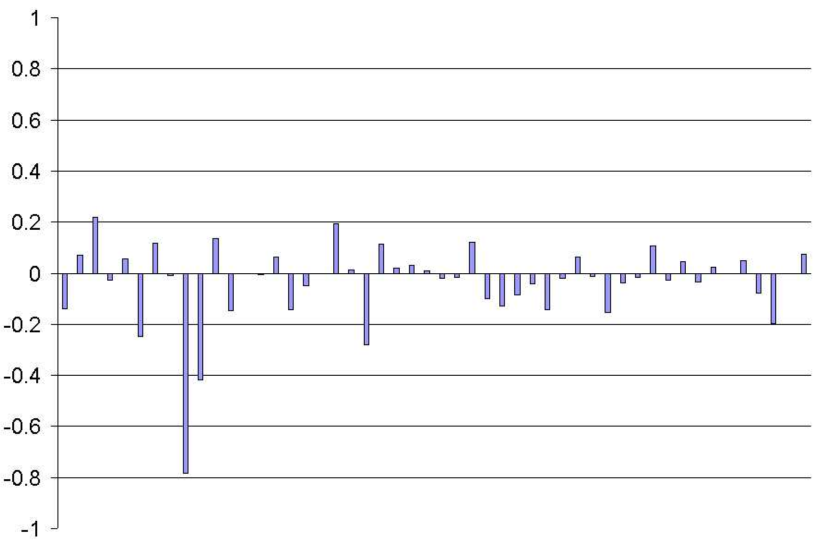

Fig. 1. Difference in Average Precision comparing our official submission and median average precision values.

We also did work on the categorization task. We worked with the full text documents on the triage task. Our approach consisted on using the features extracted by TAPIR with their respective weights. This generated a significantly large number of features (more 450k features). We tried several ways to use SVM-light with the whole feature set but results on a training validation split were not satisfactory. We the decided to perform feature selection using correlation coefficient to select the top 1000 features. These 100 features were used to reduce the vector representation of the documents in the training set so that we could use the WEKA tool kit [5] to explore other categorization methods. We tried several methods using a 10-fold validation on the training set and selected the two methods with high performance. These methods are Naive Bayes and Logistic regression. Table 1 shows our results on the 10 fold validation of the training.

Table 1. Performance of categorization methods in the training set

Method	Recall	Precision	F1	Uraw	Unorm
Naive Bayes	0.533	0.2433	0.3342	3378	0.4504
Logistic Regression	0.453	0.3184	0.3740	3036	0.4048

We then used the same 1000 selected features to represent the test set and run the two classifiers (naive Bayes and logistic regression) over the test set. Table 2 shows our results on the test set. Surprisingly the performance is quite different from the performance obtained in the training set. Although the results are still positive, we can observe a significant drop in performance for both classifiers. The naive Bayes classifier maintains a more stable performance but is still between 44% and 56% (depending on the measure used to compare the performance) below the performance obtained on the training set. We wanted to try more classifiers but unfortunately we run out of time and in fact we could not submit the results of this task on time for the official date but we show here our unofficial results.

Table 2. Performance of categorization methods in the test set

Method	Recall	Precision	F1	Uraw	Unorm
Naive Bayes	0.279	0.1408	0.1870	1612	0.1938
Logistic Regression	0.156	0.1043	0.1251	742	0.0891

3 Conclusion

In conclusion we can say that our statistical language model toolkit (TAPIR) was able to process the large collection of documents for the adhoc retrieval. Our performance is slightly below the average but we expect to improve it in the next year TREC by adding more semantic features extracted from the text and including some mechanism for pseudo-relevance feedback or local context analysis.

Our text categorization experiments using the combination of probabilistic features and correlation coefficient showed that it is possible to get positive results in the triage task. We expect to continue our work on text categorization using phrases and more complex features as well as exploring other methods for categorization.

References

1. Lavrenko, V. and Croft, W. B. Relevance-based Language Models. In *Proceedings of SIGIR'01*, pages 120–127. ACM, New York, 2001.
2. Ponte, J. M. and Croft, W. B. A language modeling approach to information retrieval. In *Proceedings of SIGIR'98*, pages 275–281. ACM, New York, 1998.
3. Srikanth, M., Ruiz, M. E. and Srihari, R. UB at TREC 12: HARD and Genomics Tracks. In *Proceedings of the twelfth Text Retrieval Conference (TREC2003).* NIST Special Publication: SP 500-255, page 751–755, 2003.
4. Cronen-Townsend, S. and Croft, W. B. Quantifying Query Ambiguity. In *Proceedings of HLT'02*, 2002
5. Witten, I. H. and Frank, E. *Data Mining: Practical machine learning tools with Java implementations.* Morgan Kaufmann, San Francisco, 2000.