Applying SIGMA to the TREC-7 Filtering Track

Grigoris J. Karakoulas

Global Analytics Group
Canadian Imperial Bank of Commerce
161 Bay St., BCE-11, P.O. Box 500,
Toronto ON, Canada M5J 2S8
karakoul@cibc.ca

Innes A. Ferguson

Active On-line Systems Ltd.
13 Grange Park, Flat 4
London W5 3PL
UK
innes@active-online.net

Abstract

This paper presents the research method and results from applying SIGMA (System of Information Gathering Market-based Agents) to the adaptive filtering task of the TREC-7 Filtering Track. Our work in SIGMA is based on the research hypothesis that a multi-agent learning approach, where each agent learns a local model for information filtering, performs better than a single-agent learning approach that approximates the whole input space with a single model. We report on experiments for testing this hypothesis as well as for constructing the feature space model for this task. This has been SIGMA's first application to TREC experiments and to an IF task over a large document collection. The system was able to scale to the complexity of the task. The performance of SIGMA in TREC-7 is encouraging for further work.

1. Introduction

There has been growing interest within the Information Retrieval and Machine Learning communities for developing methods that combine retrieval results from different learning models for text categorization and routing (see for example Belkin et al., 1993; Lewis et al., 1996; Schapire et al., 1998; Vogt et al., 1998). Such work is supported by empirical evidence showing that for the same information need different users may retrieve sets of documents with little overlap amongst them (McGill et al., 1979). In our work, we have developed SIGMA (System of Information Gathering Market-based Agents) for multi-agent incremental learning and applied it to various information filtering (IF) tasks (Karakoulas & Ferguson, 1995;1996a,b). Due to the multi-dimensionality, partial observability and non-stationarity of an IF task, a multi-agent learning approach, where each agent learns a local model, should perform better than a single-agent learning approach that approximates the whole input space with a single model.

Our interest in participating in TREC-7 (our first TREC participation) and particularly in the adaptive filtering task of the Filtering Track is for two reasons: (i) to evaluate the hypothesis that multi-agent learning with SIGMA improves IF performance over a single-agent learning model; and (ii) to test the scalability and robustness of our IF system for this complex task. In the next section we present the method for performing adaptive information filtering with SIGMA. We then describe the experiments for the Filtering Track and present some of the results. In the last section of the paper we discuss future work.

2. Adaptive Information Filtering with SIGMA

Similarly to economic markets, the computational economy of SIGMA can be defined in terms of goods and agents that produce and consume those goods. The *goods* traded in SIGMA are information items (i.e. AP news articles) in different representation forms depending on the stage of processing. The agents are of three general categories: (i) the *consumer* agents, (ii) the *producer* agents and (iii) the *broker* agents.

A consumer, Profile Selector (PS), represents a user's goal and preferences for a topic (i.e. one of the 50 TREC-7 topics) over a time period. Producers, Feature Extractors (FE) and Profile Generators (PG), transform goods from an input form into an output form according to their learning techniques. In response to a consumer's demand for goods the producers enter the local market and compete with each other to serve as efficiently as possible the demands for goods from other agents — consumers or other producers — within the market. A producer determines the price that maximizes its profit by learning how to produce goods, that a consumer is expected to buy, and estimating their demand. A broker is responsible for implementing the bidding policy for a particular consumer's demand for goods, namely setting up the auction each time a demand is posed in the market and deciding which producers win the bidding. The broker also maintains the history of the performance of the producers in serving the particular consumer's demand. The producers that succeed in the bidding sell goods to the consumer at their respective prices. The goods are of different quality. The consumer is endowed with a budget. At any time the consumer is allowed to "shop around" by probabilistically assigning its preferences among the producers and allocating its budget for buying one or more goods. Given its choices for goods these stochastic preferences are reinforced by the feedback that the consumer receives from the environment. The feedback for each good is also propagated by the consumer to the producer from which the product was bought. The market agents used in this task were built as follows.

FE Agent. It transforms an article or a topic according to the Vector Space Model (VSM). The outputs of an FE are the VSM representations of articles and document frequency (DF) tables. The VSM representation currently consists of terms only. Three different term weighting schemes were implemented: (i) simple tf-idf (ii) Salton & Buckley's (1987) tf-idf and (iii) a version of SMART similar to the one by AT&T in TREC-6 (Singhal, 1997). More specifically, (iii) is defined as:

$$A = (1 + \log(tf)) / (1 + \log(average(tf_d))) \tag{1}$$

where tf is the term frequency in document d;

$$B = \log((N + 1) / (df)) \tag{2}$$

$$C = 1 / [0.8 + 0.2 \times (\text{no. of unique words in d / avg no. of unique words per doc})] \tag{3}$$

and from (1), (2) and (3)

$$w(t, d) = A \times B \times C \tag{4}$$

In the case of a topic w(t,d) = A.

PG Agent. Upon creation it is initialized with the profile from the topic query. It retrieves VSM articles from an FE agent and determines which articles to purchase using the normalized cosine between the article and profile vectors as the similarity measure together with a similarity threshold. When the PG receives relevance feedback it updates its profile through a modified Rocchio formula:

$$P_{t+1} = \begin{cases} P_t + \beta\overrightarrow{D_j} & \text{if } D_j \text{ relevant} \\ P_t - \gamma\overrightarrow{D_j} & \text{if } D_j \text{ non-relevant} \end{cases} \quad (5)$$

where $P=[w_1,w_2,...,w_N]$. Only positive weights are allowed. New terms from relevant documents are added to the profile by multiplying their weights with a learning rate δ. The length of the profile, N, is fixed. Terms with relatively low weights can be replaced by terms with higher weights.

PS Agent. The decision task of a PS agent at each time *t* amounts to buying from the set of the PG agents that have won the bidding at time *t*, articles such that the cumulative rewards from buying articles in the long term is maximized. The rewards are the F1 and F3 utility measures based on the relevance judgements. Through such feedback the PS agent learns a policy for which PG agents to buy articles from (for more details see (Karakoulas & Ferguson, 1996b)).

3. Experimentation and Results

Each AP document was preprocessed by filtering words using a stopword list and stemming them. In addition, terms with single occurrency were periodically removed from the frequency tables so that noise due to spelling errors is reduced. Each of the topic descriptions was automatically preprocessed similarly to AP documents for forming the initial query to the system. Words in the concepts section preceded by NOT were removed. To eliminate noise we experimented with forming the topic queries using either limited number of keywords (10, 20 or 30) or specific sections of the topic descriptions. Queries based on the title and concepts gave the best performance by training on the data of 1988/02-1989/04 and testing on the data of 1989/05-1989/12. This data split was used in all the experiments reported in this section. For the performance evaluations we used the F1 and F3 TREC-7 utility measures as well as the geometric mean between precision and recall.

Our experiments can be categorized into two classes: (a) feature space modeling and (b) learning. In the first class we addressed the following questions:

(i) does performance improve by limiting the number of terms represented in each document;

(ii) does the addition of more terms from relevant documents to the initial query improve performance;

(iii) how critical is the choice of the specific method for modelling the feature space and creating the VSM representation of a document.

For (i) we experimented with 20, 50 and 100 terms as the document length. The second option gave the best results. In (ii) we found that increasing the initial query length by 20 terms through relevance feedback gave the best performance. For (iii) we compared the three weighting schemes that were incorporated in the FE agent as described in the previous section, i.e. simple tf-idf, Salton&Buckley's tf-idf and a version of SMART. The latter method gave the best performance.

For the second class of experiments we addressed the following two questions:

(iv) how does SIGMA compare with a system that has no learning capabilities and only selects documents based on the initial query

(v) since SIGMA has a two-level learning mechanism how much does SIGMA improve performance over a system that only uses our version of Rocchio's formula to learn a topic profile

For (iv) we developed an IF system that selects documents by first estimating the normalized cosine for the similarity of the document with the initial query profile and then predicting the document to be relevant if its similarity exceeds a predefined threshold. The performance of SIGMA on the testing data over all topics was on average four times better than the performance of this system that used no relevance feedback. For (v) we used SIGMA with only one PG agent. Learning only takes place in the PG agent through our version of Rocchio's formula and relevance feedback. The learning factors in (5) were set as follows: β=0.9, γ=0.1 and δ=0.2. The performance of SIGMA on the testing data over all topics was on average 85% better than the performance of this single agent with one level of learning.

SIGMA's performance with respect to the other participants in the adaptive filtering task of TREC-7 is shown in Tables 1 and 2, one for each utility run. The tables were constructed using the evaluation results from NIST. For each topic we compared the performance of SIGMA with respect to the minimum, median and maximum performance on that topic. The tables show the averages over all topics of the number of times SIGMA's performance ranked equal to the minimum (=min), less than the median (<=median), greater than the median (>median) and equal to the maximum (=max) peformance. SIGMA mostly ranked lower than the median performance in the F1 run and above the median in the F3 run. The reason for this difference in performance between the two measures is because the F3 utility measure is more biased towards recall than the F1 measure. Due to time constraints we did not tune SIGMA for precision which has been the goal of TREC.

	=min	<=median	>median	=max
1988	0.42	0.5	0.08	0
1989	0.56	0.36	0.08	0
1990	0.62	0.34	0.04	0

Table 1: Comparison of SIGMA for the F1 run.

	=min	<=median	>median	=max
1988	0.36	0.48	0.16	0
1989	0.46	0.38	0.16	0
1990	0.46	0.44	0.08	0.02

Table 2: Comparison of SIGMA for the F3 run.

4. Discussion

This has been our first participation in TREC. We were able to perform various experiments for empirically validating research hypotheses and successfully carry out the adaptive filtering task on the AP collection. This has been partly due to the scalability and robustness of SIGMA with respect to the dimensionality and complexity of the task. According to our evaluation, the performance of SIGMA ranked below but close to the median level compared to the rest of the participants in the adaptive filtering task. Given the time constraints that we faced the results are encouraging for our research in SIGMA. The comparisons with two benchmark IF systems have shown that the two-level learning approach of SIGMA can substantially boost the performance of an one-level learning approach.

There is work underway for extending the feature extraction capabilities of the FE agents with the extraction of phrases. Future work will also focus on a method for dynamically learning the similarity threshold in PG agents.

References

Belkin, N.J.; Cool, C.; Croft, W.B.; and Callan, J.P. The effect of multiple query representations on information retrieval performance. In *Proceedings of the Sixteenth International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-93)*. Association for Computing Machinery.

Karakoulas, G.; and Ferguson, I. 1995. A computational market for information navigation in multidimensional spaces. In *Proceedings of AAAI 95 Fall Symposium on AI Application in Knowledge Navigation and Retrieval*, AAAI Press.

Karakoulas, G.; and Ferguson, I. 1996a. SIGMA: Integrating learning techniques in computational markets for information filtering. In *Proceedings of AAAI 96 Spring Symposium on Machine Learning and Information Access*, AAAI Press.

Karakoulas, G.; and Ferguson, I. 1996b. A computational market model for multi-agent learning. In *Proceedings of AAAI 96 Fall Symposium on Learning Complex Behaviors in Adaptive Intelligent Systems*, AAAI Press.

Lewis, D.; Schapire, R.; Callan, J.; Papka, R. 1996. Training algorithms for linear text classifiers. In *Proceedings of the Nineteenth International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-96)*. Association for Computing Machinery.

McGill, M.; Koll, M.; and Norreault, T. 1979. An evaluation of factors affecting document ranking by information retrieval systems. Syracuse University School of Information Studies.

Salton, G.; and Buckley, C. 1987. Term weighting approaches in automated text retrieval. Technical Report 87-88, Cornell University, Department of Computer Science.

Schapire, R.; Singer, Y.; and Singhal, A. 1998. Boosting and Rocchio applied to text filtering. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-98)*. Association for Computing Machinery.

Singhal, A. 1998. AT&T at TREC-6. In *Proceedings of Sixth Text REtrieval Conference (TREC-6)*, NIST.

Vogt, C.; and Cottrell, G. 1998. Fusion via linear combination for the routing problem. In *Proceedings of Sixth Text REtrieval Conference (TREC-6)*, NIST.