© 2016 IJSRST | Volume 2 | Issue 3 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Performance Analysis of an Ontology Based Crawler Operating in a Distributed Environment

Wael A. Gab–ALLAH, Ben Bella S. Tawfik,Hamed M. Nassar
Faculty of Computers & Informatics, Suez Canal University, Ismailia, Egypt

ABSTRACT

Crawlers are being increasingly utilized to retrieve information from distributed information sources, such as the Web. We have implemented one that makes use of some novel algorithms and techniques, namely, a novel IR architecture, an efficient query expansion algorithm based on WordNet, a new crawling technique based on ontology and a new rapid filtering algorithm based on semantic similarity. The experimental results of the implemented crawler, named Ontology Based Distributed Information Retrieval (OBDIR) system, show superiority to those obtained from systems based on the standard Breadth First (BF) search technique. In this paper we analyze the performance of the OBDIR system. We develop a probabilistic model that captures the operational dimensions of the system. The model makes heavy use of Bayes' theorem and can help establish a foundational theory for DIR. We study such performance metrics as recall and precision, and allude to other performance tools such as accuracy and ROC space. The study shows that by carefully choosing the keywords the performance of the crawler is enhanced greatly.

Keywords: Information retrieval, Web search, Focused crawler, Ontology

I. INTRODUCTION

Information Retrieval (IR) is a branch of computer science that deals with automated information storage and retrieval. The objective of a text retrieval system is to find those documents in a text database that are relevant to a user's criteria. We say that a document is relevant if it answers the user's query, otherwise the document is irrelevant.

Ontology–based web crawlers are the state of the art technology for information retrieval from distributed information systems such as the Web [1]. We have proposed a set of algorithms and techniques to improve the performance of DIR, namely, a novel IR architecture, an efficient query expansion algorithm based on WordNet, a new crawling technique based on ontology, and a new rapid filtering algorithm based on semantic similarity. Our proposed crawler model is shown in Fig. 1. It is comprised of two phases. Phase I is made up of two parts, a and b. In Part a of Phase I, the query supplied by the user is expanded using the ontology vehicle of the implementation, namely, WordNet. In Part b of Phase I, the proposed crawler searches the text database, i.e. the Web, for relevant documents--- those that meet the user's query. The end result of Phase I is a collection of documents that the system thinks relevant (positive). Unfortunately, the crawler's retrieval may not be perfect based on the work of Phase I only. For example some documents may be on the borderline between relevant and irrelevant. Also, in Phase I an irrelevant document can be judged relevant based on bad semantics interpretation. Here is where Phase II comes in. A filtration process is exerted in Phase II of the proposed crawler to retain only the documents most relevant to the user's query, and discard all others.

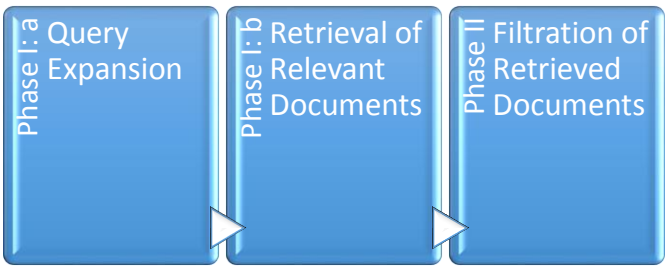

Figure 1 : The proposed crawler's architecture: two phases, with phase I made of two parts.

Given a database of documents and a user's query, we can locate those documents that meet the user's information needs. Because there is no precise definition of which documents in the database match the user's query, uncertainty is inherent in the information retrieval process. Therefore, probability theory is a natural tool for formalizing the retrieval task. In this paper, we propose a Bayesian approach to one of the conventional probabilistic information retrieval models. We discuss the motivation for such a model, describe its implementation and present some experimental results.

A crawler is supposed to retrieve only the documents that are relevant to the query and ignore those that are irrelevant. However, the crawler makes errors. Specifically, it may retrieve a document that is irrelevant (type I error), or ignore a document that is relevant (type II error.) A common abstraction in this context is to call the relevant documents positive and the irrelevant negative, and to describe the decision of the crawler to either retrieve or ignore as true or false. Thus, when the crawler ends its work, it has actually partitioned the universe of Web documents into four subsets, shown in Fig. 2: True Positive (TP), False Positive (TP), True Negative (TP) and False Negative (TP). In the names of these subsets, the second word refers to the class the crawler has placed the subset in, positive or negative, and the first word refers to our judgement of that placement, true (correct) or false (erroneous).

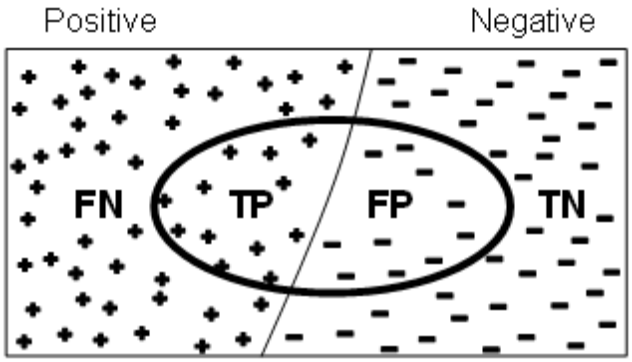

Figure 2: The original two sets of documents in the search space, positive (P) and negative (N), are partitioned by the crawler into four subsets: the two it retrieves, namely, true positive (TP), false positive (FP), and the two it leaves out, namely, false negative (FN), true negative (TN)

Referring to Fig. 2, the elements inside (outside) the ellipse denote the documents retrieved (ignored) by the crawler in the belief that they are positive (negative). Whether this classification by the crawler is correct is checked by looking at the rectangle as a whole, and that is how the adjectives true and false are written. If the element believed by the crawler to be positive (i.e. is inside the ellipse) is really positive (i.e. has the + shape), the crawler's judgement as True. Else, it is False.

II. METHODS AND MATERIAL

1. Related Work

Information retrieval is largely a probabilistic endeavor. In our case, the crawler receives the user a query whose contents are probabilistic. The way the query is expanded is probabilistic. Then during the search, whether a document will match the query or not is probabilistic, and is even more so when the matching is carried out on the semantics rather than on the syntax, which is the case in our crawler. Finally, the filtering operation is also probabilistic, due to the fact that similarity computation is mainly subjective, giving rise to its being biased in some cases in one direction and in other cases in an opposite direction.

It stands to reason than that research attempts to model IR systems depends principally on probability theory and mathematical statistics. In fact probabilistic information retrieval models date back to the early 60's, but have rarely been used in operational retrieval systems. However, the probabilistic model was perhaps the first IR model with a firm theoretical foundation.

In [2] a statistical interpretation of term specificity and its application to retrieval is introduced. The authors laid down the foundation for a statistical model capable of assessing the specify of a term and applied the established concepts to information retrieval. In [3] the authors improve the precision ratio using semantic based search. In the process they calculate the probability that the true results would supersede the false results. In [4] several probabilistic models that can be used in information retrieval are introduced. They mostly depend on probability theory and information theory. They make use of such well known concepts as conditional expectation and entropy. In [5] the author presents methods that can improve information retrieval

with textual analysis. The author relies on Bayesian models, but still suggests other related models based on statistical classification. In [6] the authors carry out analysis on the performance of mobile agents regarding their use for query retrieval. They introduce a viable probabilistic model that captures the stochastic operational elements of the retrieval system.

In [7] the authors present an interesting method for unsupervised semantic similarity computation between terms using web documents. They end their study with a probabilistic model to evaluate the overall performance of a system making use of their methodology. In [8] the authors introduce Bayesian probabilistic models tailored specifically for image retrieval. However the concepts presented there are suitable for use in the retrieval of generic documents. The commonplace concepts and metrics, such as recall, precision and accuracy, can be generalized to serve in documents of a general nature and not only in images. In [9] a number of elementary probabilistic models for information retrieval are introduced. The interesting thing about this work is that it uses accessible mathematical instruments and yet it results in accurate classification results.

2. Model

The four subsets produced by the crawler upon ending a retrieval lead to what is called a confusion matrix, shown in table 1.

Item classified as	Sample true identity	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN
	P=TP + FN	N=FP + TN

Table 1: The Confusion Matrix: the search set is partitioned into four subsets, namely TP, FP, FN and TN.

Using the confusion matrix, we can define a performance measure, called accuracy, for the crawler as follows.

$$\text{Accuracy} = \frac{\text{Number of correct classifications} = \text{TP} + \text{TN}}{\text{Total number of samples} = \text{P} + \text{N}}$$

The problem with accuracy is that it does not distinguish between the two types of errors the system makes (False Positive or False Negative). For example, two systems may obtain the same accuracy but behave quite differently on each category. If one system has 100% accuracy on one category and 41% on the other, while another system produces 70% for each category, it is hard to claim that the first system is better. As a result, overall accuracy cannot be relied upon to evaluate systems on a dataset, and Precision and Recall are used instead. Precision can be seen as a measure of exactness or fidelity, whereas Recall can be seen as a measure of completeness. Their definitions are:

$$\text{Precision} = \frac{\text{Number of True Positives} = \text{TP}}{\text{Total number of positive samples} = \text{P}}$$

$$\text{Recall} = \frac{\text{Number of True Positives} = \text{TP}}{\text{Number of Classified Positive} = \text{TP} + \text{FP}}$$

The problem with Precision and Recall is that they pay more attention to the system's ability to identify the positives, and less attention to its ability to identify the negatives.

Receiver Operating Characteristic (ROC) analysis can solve the problems of both Accuracy and Precision/Recall. A ROC graph, shown in Fig. 3, plots the True Positive Rate (TPR), on the y-axis, against the False Positive Rate (FPR), on the x-axis. TPR is defined as$\text{TPR} = \frac{\text{TP}}{\text{P}}$, same as Recall, and represents the benefits (as we want it maximized), and FPR is defined as the ratio of False Positives (which are actually Negatives) to the Negatives, i.e., $\text{FPR} = \frac{\text{FP}}{\text{N}}$, and represents the cost (as we want it minimized.)

The diagonal line in a ROC graph, from the left bottom to the right top corner, is called the random guess line, and is used to judge the whether the classification is good or bad. A good crawler is one that operates in the upper-diagonal area of a ROC graph. Points above the random guess line indicate good classification, whereas those below the line are considered as bad classification. The (0, 1) point is called a perfect classification, as it means that the crawler retrieved all the relevant documents and did not retrieve a single irrelevant document. The shorter the distance to the (0, 1) point, the better the classification, and vice versa. Thus, the farthest point, namely (1, 0), indicates the worst possible classification, as it means that of the documents the crawler retrieved, all are irrelevant and none is relevant.

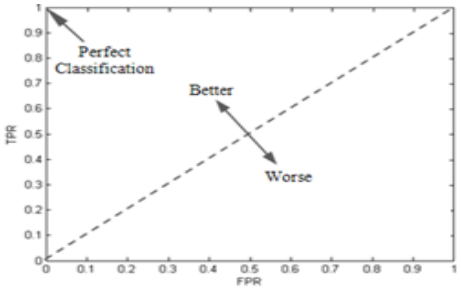

Figure 3: The ROC space graph. The diagonal dashed line, called the Random Guess Line, divides the classification area into two parts: upper, which is desirable, and lower which is undesirable.

Let P denote the event that a document selected at random from the Web is positive, and N denote the event that it is negative. Now, let p be the fraction of positive documents on the Web. Then,

Pr [P] = p

Pr [N] = 1-p

Let CP denote the event that, upon examination, the crawler classifies a document as positive, and CN the event that the crawler classifies it as negative. With a non-perfect crawler, it is possible that it will errors in the classification. We here assume that the amount of error depends whether the document is positive or negative. Specifically, we assume u to be the degree of accuracy for positivity of the crawler, i.e. the fraction of time that the crawler is accurate when handling a positive document, and v to be the degree of accuracy for negativity of the crawler, i.e. the fraction of time that the crawler is accurate when handling a negative document. That is,

Pr [CP|P] = u

Pr [CN|P] = 1-u

Pr [CN|N] = v

Pr [CP|N] = 1-v

Relating the above events and probabilities to the four subsets created after the crawler ends its search job, as mentioned earlier, is now straightforward. We can now derive the probability that a document selected at random from the searched documents belongs to any of the four mentioned subsets as follows.

Pr [TP] = Pr [CP∩P] = Pr [CP|P] Pr [P] = up

Pr [FP] = Pr [CP∩N] = Pr [CP|N] Pr [N] = (1- v) (1- p)

Pr [TN] = Pr [CN∩N] = Pr [CN|N] Pr [N] = v (1- p)

Pr [FN] = Pr [CN∩P] = Pr [CN|P] Pr [P] = (1- u) p

Using these probabilities, we can find expressions for the recall, precision and accuracy as follows.

$$\text{Recall} = \frac{\Pr[\text{TP}]}{\Pr[\text{P}]} = \frac{up}{p} = u$$

$$\text{Precision} = \frac{\Pr[\text{TP}]}{\Pr[\text{TP}] + \Pr[\text{FP}]} = \frac{up}{up + (1-v)(1-p)}$$

$$\text{Accuracy} = \frac{\Pr[\text{TP}] + \Pr[\text{TN}]}{\Pr[\text{P}] + \Pr[\text{N}]} = up + v\,(1 - p)$$

Below, we plot both the recall and precision vs. the degree of accuracy for positivity of the crawler, and the degree of accuracy for negativity of the crawler.

III. RESULT AND DISCUSSION

We notice that our model in Section 3 is built on only three variables: the probability that a sample randomly selected from the population being positive, p, the probability that the system classifies a positive sample correctly, u, and the probability that the system classifies a negative sample correctly, v. These are called a priori probabilities, i.e. probabilities that are given or known beforehand. If we have them on hand, we can find out all the crawler's performance metrics, and that will be our task in this Section.

So here is our strategy. First, we will run the system to obtain its performance metrics. Then, we will utilize them to evaluate the three a priori probabilities empirically using the formulas we developed in Section 4, in a reverse engineering style. Having obtained the a priori probabilities empirically, we can plug them in the model's formula to theoretically evaluate other performance metrics, such as the accuracy.

We start by running the OBDIR crawler system that we have designed and implemented on datasets of different sizes. Specifically, we run it on samples of 100, 250, …, 2000 documents. In each of these datasets, we make sure that 85% of the documents are relevant and 15% are irrelevant. That is, referring to the model of Section 3, we have p = 0.85. We obtain for each dataset the two metrics of the system: Recall and Precision.

Our experimental results for the Recall vs. the number of documents of the implemented OBDIR system are shown in Fig. 4. It is evident that the recall increases slightly as the number of searched documents increases. However, an average Recall value of 0.6 can be safely assumed. Thus in our model we will have u = 0.6

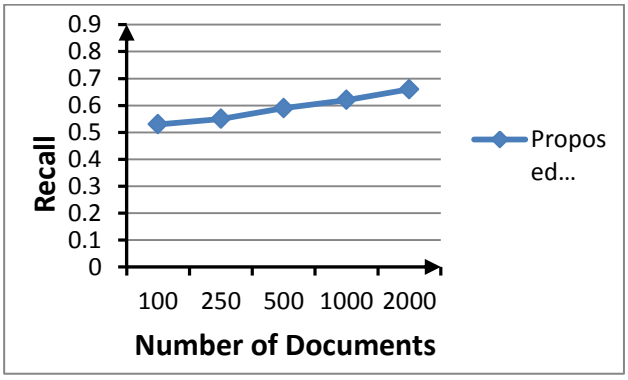

Figure 4: The Recall of the proposed system vs. the number of documents.

An average Recall of u =0.6 can safely be assumed.

Our experimental results for the Precision vs. the number of documents of the implemented OBDIR system are shown in Fig. 5. Let S denote the precision of the system. It is evident that the precision increases then stabilizes near 0.8.

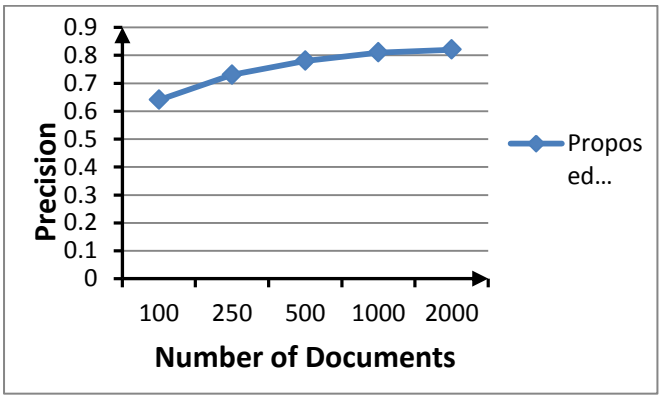

Figure 5: The Precision of the proposed system vs. the number of documents.

An average Precision S =0.8 can safely be assumed.

That is, an average precision value of 0.8 can be safely assumed, i.e. S =0.8. Referring to our model of Section 3, we have

$$S = \frac{up}{up + (1-v)(1-p)} = 0.8$$

Solving for v we get

$$v = 1 - \frac{(1-S)up}{S(1-p)}$$

Substituting p =0.85, u =0.6 and S =0.8, we get

$$v = 1 - \frac{0.2 \times 0.6 \times 0.85}{0.8 \times 0.15} = 0.15$$

That is, our system has u =0.6 and v =0.15, which means it is more (actually four times) accurate to classify a positive item than to classify a negative item.

Now, using our probabilistic model of Section 3, we will study the effect of the probabilities u and v on the system's performance. First, we evaluate how both the accuracy and precision change as the probability, u, of correctly classifying a relevant (positive) document changes, for a fixed v=0.15. This is shown in Fig. 6. Then, we evaluate how both the accuracy and precision change as the probability, v, of correctly classifying an irrelevant (negative) document changes, for a fixed u=0.6. This is shown in Fig. 7.

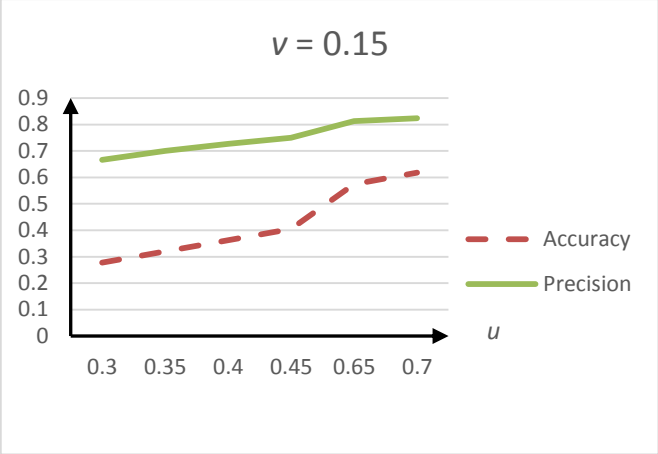

Figure 6: The precision and accuracy of the proposed system

Vs. the probability of correctly classifying a relevant (positive) document, u.

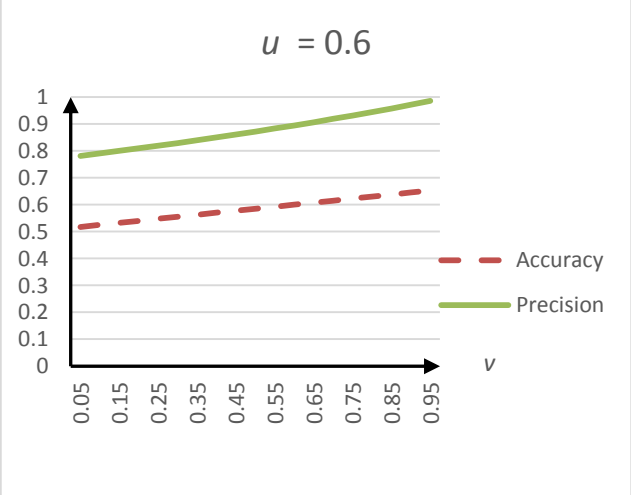

Figure 7: The precision and accuracy of the proposed system

Vs. the probability of correctly classifying an irrelevant (positive) document, v. From the two Figures, Fig 6 and Fig 7, we can easily see that the effect of increasing v is more profitable, especially for the precision of the system. Actually, as we see in Fig. 7, the precision almost reaches 1 as v approaches 0.9. But we notice that the increase of v is less profitable for accuracy. From

Fig. 7 again, we see that the accuracy is almost unaffected by the increase in *v*.

IV. CONCLUSION

In this paper, we have introduced a probabilistic Bayesian model for a DIR system that we have designed and implemented, under the name Ontology Based Distributed Information Retrieval (OBDIR). We present experimental results including well know retrieval metrics, such as accuracy, recall and precision. The results prove the viability of the model and can be used as a base for future retrieval theory. Using the proposed Bayesian model, we were able to gain some insights about our OBDIR system. For example, we were able to realize that the system's precision can improve greatly if we can increase its ability to correctly recognize irrelevant documents, *v*. But at the same time, we have found that increasing this ability will not increase the system's accuracy. This tells that, based on which one is of more importance to the user, this ability may or may not be increased.

V. REFERENCES

[1] S.SASIREGA, A.Jeyachristy, (2014). "Ontology Based Web Crawler for Mining Services Information Retrieval". International Journal of Computer Science and Mobile Computing, Vol. 3, No. 11, pp.325–330.

[2] Jones, K. (2004). "A Statistical Interpretation of Term Specificity and its Application to Retrieval". Journal of Documentation, Vol. 60, No. 5, pp. 493-502.

[3] Amudaria, S., and S. Sasirekha, (2011). "Improving the precision ratio using semantic based search". Signal Processing, Communication, Computing and Networking Technologies (ICSCCN), International Conference on. IEEE, pp. 465–470.

[4] Fuhr, N. (1992). "Probabilistic models in information retrieval". The Computer Journal, Vol. 35, No. 3, pp. 243–255

[5] Teevan, J. B. (2001). Improving information retrieval with textual analysis: Bayesian models and beyond (Doctoral dissertation, Massachusetts Institute of Technology).

[6] Selamat, A. and M. H. Selamat, (2005). "Analysis on the Performance of Mobile Agents for Query Retrieval", Information Sciences, Vol. 172, No. 3, pp: 281–307.

[7] Iosif, E. & Potamianos, A. (2010). "Unsupervised semantic similarity computation between terms using web documents". IEEE Transactions on Knowledge and Data Engineering, Vol. 22, No. 11, pp. 1637–1647.

[8] Stathopoulos, V., & Jose, J. M. (2011). "Bayesian Probabilistic Models for Image Retrieval". WAPA, pp. 41–47.

[9] Lavrenko, V. (2010). "Introduction to Probabilistic Models for Information Retrieval," 33rd International ACM SIGIR conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, pp. 905-