Evidence Accumulation with Competition in Information Retrieval

Aurelio López-López
Electronics
I.N.A.O.E.
P.O.Box 51 and 216
Puebla Pue. 72000, México
allopez@gisc1.inaoep.mx

Sung H. Myaeng
Department of Computer Science
Chungnam National University
Taejun, Korea
shmyaeng@cs.chungnam.ac.kr

Abstract

A basic component of a retrieval system is the process of comparing a representation of a document and that of an information need. In this paper, we take the view that information retrieval can be modeled as an inference process and proposed a scheme that associatively retrieves potentially relevant documents accumulating evidence from various sources. This scheme that can serve as a precision-improving technique proceeds as follows: it 1) selects an initial subset of promising documents, 2) constructs a network of those selected documents based on their similarities so that they can compete each other, and 3) accumulate pieces of evidence that a document is relevant, based on multiple criteria so that the documents are ranked. The criteria, by which individual document are evaluated, are currently pragmatic (e.g. importance of terms for the user), semantic (e.g. relationship of terms in query and document), and salience (e.g. term importance in documents). This scheme models the phenomenon of competition among documents for their relevance to a given query. Our experimental results show that this scheme is promising in improving retrieval effectiveness.

1. Introduction

A typical information retrieval (IR) system works with representations of information needs and documents, rather than the actual needs and documents themselves. Since there must be a gap between the representations and the actual user needs or full documents, from which the representations are derived, the retrieval task is performed under uncertainties and is prone to errors. When an exact matching does not provide exact results in a search space, a certain level of inference would be helpful. This article addresses a retrieval scheme that models IR as a process of gathering pieces of evidence that a document is relevant to the query.

The existence of the gap between the real information need and the actual query submitted to an information retrieval system was first acknowledged by Taylor [14] in his study on question negotiation. He identifies four levels of need on which the user develops his/her question when approaching a librarian. These levels go from visceral to compromised need, and a negotiation process is necessary to help the user arrive to the last level.

The process of representing document content, also known as indexing, has unresolved issues too. With the current state of text processing technology, indexing or text analysis techniques are far from being complete for IR purposes. For example, polysemy and homonymy as well as synonymy present problems in generating unambiguous and controlled representations.

The notion that IR process requires inference was put forth by a few researchers. For example, about a decade ago van Risjbergen [17] argued that a plausible inference scheme is appropriate to model information retrieval, given that documents are determined to be relevant on the grounds of a superficial description of contents. Croft and his colleagues [4,5] have been successful in modeling IR as a Bayesian inference process by representing information needs and documents within an inference net.

Taking the same stance that IR is a process that requires a certain level of inference, we present a scheme by which pieces of evidence for relevance of a document are accumulated from different sources. The scheme is based on the previously proposed model for analogical reasoning [15] where semantic similarities between predicates and their arguments are computed by representing them as nodes in a network. The IR scheme we propose here evaluates individual documents by accumulating evidence based on pragmatic (importance of terms for the user), semantic (relationship between terms in query and document), and salience (term importance in document) criteria. It also takes into account the notion of "competition" by which similar documents mutually affect each other when their relevance is computed.

Our goal in this paper is to describe the new retrieval scheme in detail and show that it can improve effectiveness in a more or less conventional retrieval setting, although the scheme has a potential to effectively incorporate richer representations of documents and information needs in a principled way [8,9]. In section 2, we further discuss additional related research. Section 3 presents the proposed scheme in detail, and section 4 reports our experiments to test the efficacy of the scheme. In section 5, we draw a conclusion and discuss future research.

2. Related research

The proposed IR scheme discussed in this paper is directly related to the ARCS (Analog Retrieval by Constraint Satisfaction) model [15] developed for analogical reasoning as mentioned above. In the original model, the goal is to retrieve analog structures comprised of predicates and arguments. Once a set of candidates predicates are identified, a network is formed with the predicates and their arguments represented as nodes and interconnected based on their similarities. This network of predicates and arguments is “executed" so that a winner node is eventually selected. Here the execution means re-calculating several times the activation levels on the nodes by combining activation levels of other linked nodes. The process is considered a form of unsupervised learning process.

Networks structures have been used for modeling information retrieval processes. Croft and Turtle [5] provide a summary of the various uses of networks in Information Retrieval. Unlike the scheme proposed here, a network for spreading activation [3,4] assumes a whole inter-connection of domain concepts and documents. Another network-based retrieval model, proposed by [5,16], also approaches information retrieval as an evidential process but uses inference networks to express it. The inference network approach is a probabilistic inference technique that accordingly incorporates and extends previous probabilistic retrieval methods. Documents are retrieved based on approximation of Bayesian inference that computes the belief that a document supports a given information need. Like spreading activation networks, inference networks also assume a complete document network for the collection.

Localist networks, i.e. nets in which each node stands for an entire concept (in contrast to a distributed representation of concepts [7]), have been explored in conjunction with information retrieval-related problems. For instance, [18] reported some experiments using a localist network to simulate the keyword selection made by human intermediaries when translating information needs to queries.

Belew [1] was one of the first to use a connectionist model in information retrieval. His contribution was to used a learning rule as part of what he called *adaptive information retrieval* (AIR), achieving automatic weight adjustment and adaptiveness to document structure and user's behavior. A brief introduction to concepts and terminology of connectionist models can be found in [6], as well as a review of applications of connectionist models to information retrieval.

3. The retrieval scheme with evidence accumulation and competition

The proposed scheme is supposed to serve as a precision-improving device that, in general terms, proceeds as follows: select an initial subset of promising documents, analyze each document looking for evidence that is similar to the query expressing similarities in a network framework, evaluate subset-wide each document by accumulating the pieces of evidence, which are generated by applying multiple criteria, and finally rank documents according to their accumulated evidence. We now describe the details of each step in the retrieval process.

While this scheme has a potential to effectively exploit richer representations of user needs and document contents, our current work simply assumes what is available in a conventional information retrieval setting. In other words, documents and information needs are both represented with index terms and their weights.

Our goal is to describe how the new retrieval scheme operates in such a conventional setting and show that the scheme alone, without any refinement of representations, can improve retrieval effectiveness.

Since the retrieval scheme is relatively expensive from a computational point of view, it should be implemented as a backend part of a retrieval system. As such, our current experimental system includes a conventional retrieval engine based on the vector space model [11] that produces a subset of the document collection, consisting of potentially relevant documents. It should be noted that the proposed model could be combined with any well-known retrieval models. Since the purpose of this step is to select a subset of the collection for the later process, furthermore, it can include various methods aimed at increasing recall such as term expansion [11].

3.1. Network representation

Once a subset of potentially relevant documents has been selected, the next step is to construct a network of the documents with the query terms appearing in them. The network consists of a number of tree-like subnets, each of which corresponds to a document. A subnet has two types of nodes: a *head node* where evidence for relevance is accumulated for the particular document it represents, and a set of *evidential nodes*, each of which corresponds to a query term and is connected to the head node. On each node is a value indicating the activation level, expressing the degree of evidence that the document is relevant to the query. In other words, each evidential node provides a piece of evidence that the head node is relevant.

All evidential nodes representing the same query term are linked each other across document subnets so that the "competition" phenomenon can be modeled. In other words, the existence of a link between two evidential nodes reduces the evidence level of the connected documents. When a query-matching term in a document is also found in another document, the degree to which the term contributes to the relevance of each document is reduced. This reduction of evidence is similar to the idea of using inverse document frequency (IDF) in document weighting schemes, in the sense that terms appearing in many documents (or subnets) are not considered as important as those appearing in a smaller number of documents. But the "competition" in the proposed network representation is not among all the documents in the collection but limited to those documents that are potentially relevant to the given query. The competition also depends on the particular query term importance determined by the user, and includes the relationship between terms in query and documents.

An example of a simple network built for three documents and a hypothetical query is illustrated in Figure 1 where an indication of competing evidence across document subnets is also shown. Three head nodes are defined as the root of each subnet, one for each of the documents, and 7 evidential nodes are connected to one of the head nodes. Each evidential node indicates that it supports the relevance of its head node representing a document, and each document's relevance is later determined by the associated evidential nodes. The dashed lines represent a link between competing evidential nodes. It should be noted that the numbers by the terms represent weights in the query or documents but do not appear in the network yet. For the sake of simplicity, activation levels are not shown in the figure.

Query: commun 0.9 system 0.6 process 0.7

Document 1: commun 0.7 system 0.8

Document 2: system 0.9 process 0.8

Document 3: commun 0.5 system 0.9 process 0.8

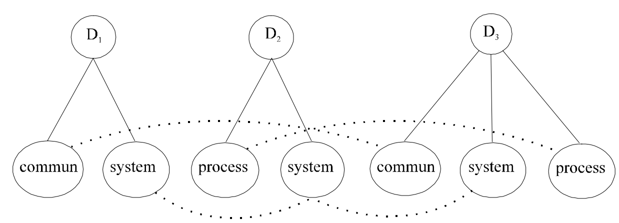

Figure 1. An example of network with competing evidence across subnets.

3.2. Evaluation of evidence

After the skeleton of the network is constructed, the next step is to evaluate each piece of evidence represented as an activation level on an evidential node so that all the evidence is accumulated later for each document. A subnet without an activation level simply indicates that a document is potentially relevant because of the associated terms in the evidential nodes, but there is no indication of how important or reliable each piece of evidence is.

The evaluation is done by employing an algorithm that applies semantic, pragmatic and salience criteria to each of the evidential nodes. That is, the value of each evidential node, which will be accumulated for each document at a later stage, is determined by the three criteria. Figure 2 depicts the notion of evaluation of an evidential node by the three criteria represented as the three boxes called evaluators that work as sources of evidence.

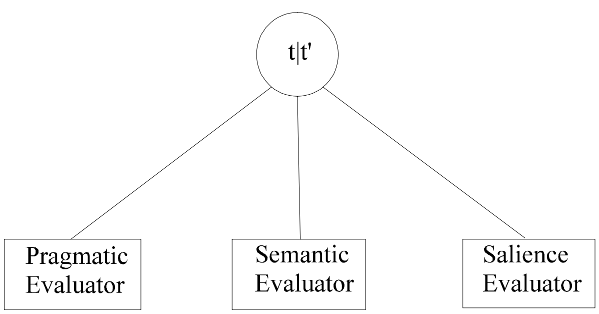

Figure 2. Evaluation of evidential node with three criteria.

While the evaluator can be implemented in a number of different ways, depending on the kinds of information available in query and document representations, we limit ourselves to more or less conventional information retrieval setting. More specifically, we interpret the semantic criterion as the relationship occurring among terms (e.g. exactly matching terms, synonymous terms, superordinates, etc.), the pragmatic criterion as the user-supplied term importance weights expressed in the query, and the salience criterion as the term importance in the document. Although this interpretation of the three criteria does not supply new information, it models the use of term relationships often found in a thesaurus, query weights, and terms weights in documents from different perspectives, making it possible to control and measure their independent contributions.

3.3. Accumulation of evidence

Given a network of documents (head nodes) and related terms (evidential nodes) together with the evaluator nodes, the relevance of each document for a given query must be calculated. This calculation is characterized with the following process of accumulating pieces of evidence for each head node, that come from connected evidential nodes.

- *Initialization phase*

  Activation levels for all the nodes and weights for the links are set based on the initial retrieval results and the parameters determined by the environment.

- *Execution phase*

  Activation levels of nodes are modified by applying a formula a number of times. As part of this phase, the current activation levels of nodes and the weights on the links between evidential nodes and the weights on the evaluator links are used to calculate new activation levels of nodes.

At the initialization phase, a maximum weight is assigned to every link, and an initial level of activation is set for every head and evidential node. The initial weights of the links leaving the evaluators are modified according to the degree of evidence the evaluators will supply to the evidential nodes. The preliminary initial weight of a link leaving an evaluator is multiplied by:

° the weight expressing the query importance of the term (represented by the evidential node) if the link is connected to the *pragmatic* evaluator,

° the weight reflecting the similarity between terms (i.e. synonymy, subordinate, exact match, etc.) if the link is connected to the *semantic* evaluator, and

° the weight indicating the importance of the term in the document if the link is connected to the *salience* evaluator.

The links established between evidential nodes (representing the same query term) of different subnets are modified negatively introducing a competing scheme through inhibiting (negative) links as mentioned above. This competing scheme produces a context-sensitive multiple-winners-take-all behavior [6] that benefits the activation of those head nodes having several supporting evidential nodes, and hopefully representing relevant documents.

The determination of the inhibiting weights for links between evidential nodes originated from the same query term is based on query term importance. This weight is lower for important terms and higher for less important ones, i.e. *1-w* if *w* is a weight between 0 and 1, representing a query term weight provided by the user. The rationale is that if the user deems a term important, there should be less competition allowing the retrieval of nearly all documents satisfying that term. On the other hand, if a term is not considered very important, the competition must be stronger causing the retrieval of those documents in which the term is highly important. Part of the network displayed in Figure 1 is included in Figure 3 showing some of the values attenuating the initial weight associated with links. Where weights are omitted, an initial maximum weight (full support between the two nodes connected by the link) is assumed for that link.

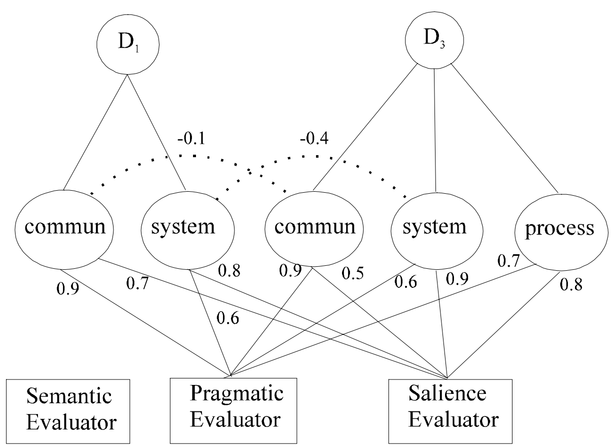

Figure 3. Partial Network with Attenuation Weights.

A "relaxation algorithm" gathers pieces of evidence from evidential nodes to head nodes, until either it has settled on a stable state (i.e. there is no significant change in any of the nodes), or a predefined number of cycles has been performed. A cycle of the algorithm calculates for each node a new level of activation based on:

- ° its current level of activation,
- ° the level of activation of those nodes supporting it (i.e. with a positively-weighted link to the node), and
- ° the level of activation of those nodes competing with it (i.e. with negatively-weighted link to the node).

This is done with the active participation of the weights associated to the links.

A formula [15] by which a new activation level at time t+1 is calculated is:

$$a_i(t+1) = a_i(t) \cdot (1-d) + (max - a_i(t)) \cdot \sum_j w_{ij} \cdot \tilde{a}_j(t) + (a_i(t) - min) \cdot \sum_j w_{ij} \cdot \underline{a}_j(t)$$

where:

t : time parameter used to distinguish values between cycles,

w_{ij} : weight associated to the link connecting node i and node j,

d : decay value controlling the settling time,

$\tilde{a}_j$: denotes those nodes supporting a_i ($w_{ij} > 0$),

$\underline{a}_j$: denotes the nodes competing with a_i ($w_{ij} < 0$), and

$max = 1$ and $min = -1$.

Once the network has settled after applying the relaxation algorithm, the head nodes can be sorted according to their final activation level, leading in this way to the ranking of the corresponding documents.

In term of computational cost, the main expense comes from the net construction that largely depends on the number of documents in the initial subset. The relaxation algorithm is linear with respect to the number of nodes in the network, where its constant is determined by the predefined number of cycles.

In contrast to vector model that only uses term weights combined in an inner product and magnitudes of query and document vectors, the proposed scheme considers term weights in query and document separately. This, together with competition links contrasting document subnets, causes that a document with strong resemblance to the query supplies evidence to documents with similar features but affects negatively those with less clear similarity.

4. Experiments

Some experiments have been performed to assess the effectiveness of the proposed retrieval scheme. A test collection widely used for Information Retrieval research was used to compare the effectiveness of the proposed scheme against the vector model [11] as a baseline retrieval method.

The experimental collection was the set of surrogates describing articles published in the Communications of the ACM from 1958 to 1979. This collection consists of 3204 surrogates and 64 information request statements.

A complete surrogate contains fields for author, title, abstract, manually-assigned keywords, Computing Reviews categories, and citation information. Only 48 of the requests have actually associated one or more documents judged relevant in the collection.

An automatic indexing scheme, based on that suggested by [11], was used for the collection. The part-of-speech tagged text of titles and abstracts was scanned for nouns, adjectives, adverbs, verbs, and foreign words. Words beginning with non-alphabetic characters were ignored as well as a list of high frequency function words with poor discriminating values. The resulting word set was expanded with individual words extracted from the keyword field present in some of the article surrogates. The supplementary words were subject to the same criteria that those from titles and abstracts regarding non-alphabetic symbols and the stop list. After the candidate indexing words were selected, they were processed with a suffix removal algorithm reducing them to word stems. This algorithm reduces different related words (e.g. 'analysis', 'analyzing', etc.) to a common stem (e.g. 'analy') as a way to produce high frequency stems from lower frequency words to achieve higher recall. The algorithm implemented [10] removes approximately 54 different types of suffixes in 5 steps.

The standard *tf* (term frequency)/*idf* (inverse document frequency) was used to calculate term weights for documents once the indexing vocabulary was extracted.

A set of 48 information need statements was processed to get the corresponding queries. They were tagged and processed in a way similar to the processing of titles and abstracts. Next a subjective evaluation of term importance was performed (based as much as possible on term frequencies in statements) leading to the assignment of weights to the query terms.

The vector model [11] was implemented as the baseline retrieval method. This model assumes that documents are identified by a collection of terms, where each term has associated a weight or importance value in each document. In this way, a collection can be conceptualized as a matrix where each row corresponds to a document, each column to a term, and the value in the matrix expresses the importance of the term in the document. Consequently, a query can be similarly associated with a vector. The similarity between query and a document is quantified by the cosine measure that calculates the cosine of the angle between query vector and document vectors in the multidimensional space. The expression for the cosine is:

$$cos(Q, D) = Q \cdot D / |Q| \cdot |D|$$

where $Q \cdot D$ is the inner product and $|Q| = \sqrt{(Q \cdot Q)}$.

The parameters governing the relaxation algorithm implementing the third step of the proposed method were similar to those used in [15]. A value of 0.01 was used to initialize network nodes as well as link weights. The decay value was set to 0.04. Evaluators had a constant activation level of 1.

Evidential nodes built for terms were evaluated by pragmatic and salience criteria only since no semantic relationship information between terms was available. The weights associated with the terms in queries determined the degree of evidence obtained from the pragmatic evaluator. Whereas, the weights attached to terms in documents affected the degree of evidence received from the salience evaluator.

Both retrieval methods used the same weights associated to the terms of the 48 queries. The set of documents produced by vector retrieval was further processed using the proposed method to obtain a new ranking.

For each query, two recall-precision tables were generated as the output of the retrieval process: one for vector retrieval, and the other for the proposed scheme. Later, retrieval outputs were further processed to calculate average precision at standard recall levels for the 48 queries. No threshold value was used with the vector retrieval; i.e. all the documents retrievable by the terms in the queries were used for the experiments. The reason was to maximize recall since the proposed scheme aims for precision improvement. Precision was interpolated at ten standard recall levels and averaged over the queries to obtain tables of average precision values at standard recall levels. Statistical tests were performed to evaluate the significance of the changes between vector retrieval and the evidence accumulation scheme in terms of precision at standard recall levels for the 48 queries. The test was a one-tailed Wilcoxon matched-pairs signed ranks test [13].

Figure 5 depicts the experimental results comparing the standard vector model and the proposed scheme in terms of average precision values at 10 recall points. Table 1 includes the numeric values of average precision at ten standard recall values when retrieval was done for the 48 queries. Percentage of change from vector retrieval to the proposed model is also shown.

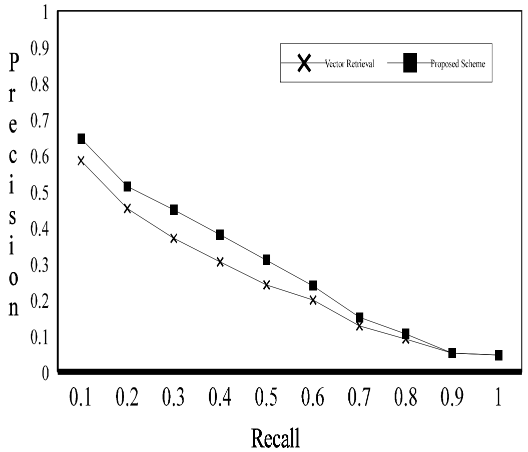

Figure 5. Average precision at standard recall levels for 48 queries.

Recall	**Precision Vector Model**	**Precision Proposed Scheme**	**% of change**
10.00	58.48	64.55	+10.38
20.00	45.25	51.36	+13.50
30.00	36.98	44.94	+21.53
40.00	30.46	38.01	+24.79
50.00	24.11	30.92	+28.25
60.00	19.88	23.85	+19.97
70.00	12.74	15.1	+18.52
80.00	9.11	10.49	+15.15
90.00	5.15	5.23	+ 1.55
100.00	4.62	4.64	+ 0.43
Average	24.68	28.91	+17.14

Table 1. Average precision at standard recall levels with percentage of change.

	Favoring			
Precision at Recall	Vector Retrieval	Proposed Scheme	Tied	One-side Probability
20.0	17	24	7	0.0548
30.0	11	29	8	0.0049**
40.0	15	29	3	0.0069**
50.0	15	28	3	0.0032**
50.0	16	25	3	0.0307*
70.0	14	22	1	0.0294*
80.0	10	15	1	0.0708
90.0	6	7	1	0.3783

* significant with p-value < $\alpha = 0.05$
** highly significant with p-value < $\alpha = 0.01$

Table 2. P-values of precision differences between the two methods.

A significance test was done at all the standard recall levels for all the queries for an overall comparison of precision values. When testing this combined evaluation, the null hypothesis that the vector retrieval performs at least as well as the proposed retrieval scheme was rejected with p-values equal to zero to four decimal places. Significance levels found at eight central standard recall values across the 48 queries are shown in Table 2. This table also includes the number of differences favoring each of the models and ties. Extreme recall values are ignored given that they are less reliable.

	Normalized Recall			**Normalized Precision**		
Query id #	**Vector**	**Proposed**	**%**	**Vector**	**Proposed**	**%**
1	96.22	95.69	-0.55	58.43	65.70	12.44
3	80.96	82.17	1.49	51.11	58.52	14.51
4	71.89	72.09	0.28	47.45	44.49	-6.24
5	98.47	98.44	-0.04	70.99	67.68	-4.67
6	99.66	99.85	0.20	79.15	92.26	16.57
7	75.93	76.50	0.76	63.05	65.54	3.95
9	97.54	99.33	1.83	79.93	87.53	9.51
10	64.91	62.82	-3.21	56.72	48.11	-15.18
11	82.68	82.84	0.20	70.45	71.18	1.04
12	99.56	98.97	-0.58	81.19	74.82	-7.85

Query id #	Normalized Recall Vector	Normalized Recall Proposed	%	Normalized Precision Vector	Normalized Precision Proposed	%
13	63.45	63.51	0.10	62.06	62.92	1.39
14	65.43	65.47	0.06	63.98	62.23	-2.74
15	79.44	79.61	0.22	65.60	68.83	4.93
16	68.86	69.05	0.28	50.44	51.53	2.16
17	49.49	49.56	0.14	40.22	41.36	2.83
18	88.79	88.25	-0.60	59.33	57.89	-2.43
19	72.58	72.56	-0.03	69.13	67.67	-2.11
20	66.10	66.51	0.61	42.18	56.54	34.03
22	63.74	63.98	0.37	53.29	57.10	7.14
23	99.14	99.55	0.41	67.36	74.23	10.21
24	30.54	30.61	0.24	28.92	29.09	0.59
25	73.72	74.05	0.45	60.94	63.04	3.46
26	87.94	88.84	1.03	72.44	81.25	12.16
27	80.43	80.91	0.60	66.63	70.18	5.32
28	79.98	80.00	0.02	79.88	81.83	2.44
29	73.60	73.42	-0.24	73.93	69.48	-6.01
30	73.98	74.52	0.72	51.83	64.19	23.84
31	99.92	99.88	-0.05	86.96	82.47	-5.16
32	99.72	99.61	-0.10	83.19	83.78	0.70
33	99.75	99.84	0.09	72.78	77.80	6.90
36	76.51	78.01	1.96	54.57	59.90	9.78
37	79.41	79.31	-0.13	49.59	49.83	0.49
38	74.83	74.90	0.09	74.64	74.51	-0.17
39	82.96	82.89	-0.08	70.83	70.67	-0.22
40	99.49	99.48	-0.01	86.62	89.20	2.99
42	46.46	46.42	-0.10	34.38	33.88	-1.45
43	57.93	59.26	2.29	37.97	46.56	22.61
44	51.30	51.52	0.44	35.94	35.10	-2.35
45	75.09	75.26	0.23	62.37	64.29	3.07
48	72.79	72.28	-0.70	45.56	42.75	-6.17
49	73.40	73.59	0.25	50.45	52.90	4.87
58	64.58	65.14	0.88	52.41	55.73	6.34
59	88.11	87.80	-0.36	72.00	71.82	-0.25
60	86.10	87.46	1.59	63.37	72.96	15.14
61	85.80	86.03	0.27	72.00	75.07	4.26
62	86.64	85.87	-0.89	63.32	53.06	-16.22
63	82.89	83.01	0.14	69.39	72.81	4.93
64	100.0	100.00	0.00	100.0	100.00	0.00
Average	78.51	78.68	0.22	62.6	64.59	3.18

Table 3. Changes in normalized measures from vector model to proposed scheme.

To get a more reliable assessment of the overall changes in ranking produced by the proposed scheme, an additional effectiveness evaluation was made using normalized recall and normalized precision [12]. These measures quantify the effectiveness of the ranking in relation to the ideal best and worst ranking. In particular, normalized recall is sensitive to the rank assigned to the last relevant document, i.e. to how long it takes to reach the last relevant document, and normalized precision is responsive to the rank of the first relevant document, i.e. how fast the first relevant document is reached.

The percentage of change in normalized measures when performing retrieval using vector model and the proposed scheme are presented in Table 3. This table contains columns for both normalized measures along with their percentage of change. These differences were also statistically tested. Table 4 summarizes the results of these tests. It includes the significance levels at which the null hypothesis (that vector model performs better or equal than the proposed scheme) was rejected when comparing the differences in normalized measures. This table displays also the number of differences favoring each of the models and ties for each measure.

Differences In	Favoring Vector Retrieval	Favoring Proposed Scheme	Tied	One-side Probability
Normalized Recall	16	31	1	0.008**
Normalized Precision	16	31	1	0.0047**

** highly significant with p-value < $\alpha = 0.01$

Table 4. P-values of differences in normalized measures from the two methods.

5. Conclusions

We have introduced a new retrieval scheme where relevance values of initially retrieved documents are computed again with the method of accumulating evidence coming from various sources. The scheme provides a way to combine evidence generated by semantic, pragmatic, and salience criteria and effectively models the notion of “competition” among potentially relevant documents.

According to the experiments performed, the proposed scheme achieved significant improvements over the vector model. Based on the 10 point average precision, the proposed scheme showed about 17%

improvement over our implementation of the vector model. When evaluating rankings using normalized measures, the scheme also showed to rank relevant documents significantly better than vector model. In the experiments done with the CACM collection with 48 queries, the proposed scheme used no additional information compared to vector model. For the pragmatic and salience criteria in the scheme, query weights and document term weights were used while the semantic criterion was not employed.

The scheme introduced in this paper provides a more rational approach to retrieval than previous models. Since the scheme approaches retrieval as an evidence-gathering process, it can be characterized as a plausible inference scheme in the direction suggested by [17]. At the same time, the approach has some similarities with the way human beings retrieve information [15].

References

[1] R.K. Belew. Adaptive Information Retrieval: Using a Connectionist Representation to Retrieve and Learn about Documents, in *Procs. ACM-SIGIR 12th Annual Conference on Research and Development in Information Retrieval*, Belkin N.J. and van Rijsbergen C.J. (Eds.), ACM Inc, New York NY, 11-20, 1989.

[2] N.J. Belkin, R.N. Oddy, and H.M. Brooks. ASK for information retrieval: Part I: Background and theory. Part II: Results of a design study, *Journal of Documentation*, 38:61-71, 145-164, 1982.

[3] P.R. Cohen and R. Kjeldsen. Information Retrieval by Constrained Spreading Activation in Semantic Networks, *Information Processing and Management,* 23(2):255-268, 1987.

[4] W.B. Croft, T.J. Lucia, J. Cringean, and P. Willett. Retrieving Documents by Plausible Inference: An Experimental Study, *Information Processing and Management,* 25(6):599-614, 1989.

[5] W.B. Croft and H.R. Turtle. Text Retrieval and Inference, in *Text-based Intelligent Systems,* P.S. Jacobs (Ed.), LEA, 1992.

[6] Doszkocs T.E., Reggia J. and Lin X. (1990), Connectionist Models and Information Retrieval, in *Annual Review of Information Science and Technology (ARIST),* Vol. 25, M.E. Williams (Ed.), Elsevier Science Publishers, pp. 209-260.

[7] J.A. Feldman. Connectionist Representation of Concepts, in *Connectionism in Perspective*, Pfeifer R., Schreter Z. (Eds.), North Holland, 25-45, 1989.

[8] A. López-López. Beyond Topicality: Exploiting the Metadiscourse of Abstracts to Retrieve Documents with Analogous Features, Syracuse University, Unpublished Ph. D. Dissertation, August 1995.

[9] A. López-López and S.H. Myaeng. Extending the Capabilities of Retrieval Systems by a Two-level Representation of Content, In *Procs. 1st Australian Document Computing Symposium,* Part I, 15-20, 1996.

[10] M.F. Porter. An Algorithm for Suffix Stripping, in *New Models in Probabilistic Retrieval,* Van Rijsbergen C.J., Robertson S.E., and Porter M.F. (Eds.), British Library R&D Report No. 5587, Computer Laboratory, University of Cambridge, Chapter 6, 1980.

[11] G. Salton. *Automatic Text Processing,* Addison Wesley, Reading, MA, 1989.

[12] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*, McGraw Hill, New York, NY, 1983.

[13] S. Siegel. *Non-parametric Statistics for the Behavioral Sciences,* Mc Graw Hill, New York, NY, 1956.

[14] R.S. Taylor. *Question-Negotiation and Information Seeking in Libraries*, College and Research Libraries, 29:178-189, 1968.

[15] P. Thagard, K.J. Holyoak, G. Nelson, D. Gochfeld. Analog Retrieval by Constraint Satisfaction, *Artificial Intelligence*, 46:259-310, 1990.

[16] H. Turtle and W.B. Croft. Evaluation of an Inference Network-Based Retrieval Model, *ACM Trans. on Information Systems*, 9(3):187-222, July 1991.

[17] C.J. Van Rijsbergen. A non-classical logic for Information Retrieval, *The Computer Journal,* 29(6):481-485, 1986.

[18] M. Wettler and R. Rapp. A Connectionist System to Simulate Lexical Decisions, in *Connectionism in Perspective*, Pfeifer R., Schreter Z. (Eds.), North Holland, 463-469, 1989.