A Technique of Data Fusion for Effective Text Retrieval

Manjusha Sanke
Associate Professor, Dept.of Information Technology
Shree Rayeshwar Institute of Engineering & Information Technology
Shiroda,Goa-India

ABSTRACT
The goal of Information retrieval systems is to provide useful information for user's information need. For a collection of documents and a given query, an IR system returns a ranked list of documents. Different IR systems based on IR models such as Vector Space, Smart Vector Space, Extended Boolean, Latent Semantic Indexing etc. produce different text documents for the same query. They rarely return the same documents in response to the same queries. This has led to the field of "data fusion", which seeks to improve the quality of results being presented to user, by combining the outputs of multiple IR algorithms or systems into a single result set. CombMNZ is a score-based fusion algorithm which adds all the reported scores for a document and multiplies the sum value to the number of retrieval models that have returned that document. This paper focuses on Norm_CombMNZ algorithm which normalizes the result obtained from CombMNZ, so that scores lie in 0 to 1 common range and better ranking judgment can be made. The performance of individual IR system is compared with the performance of data fusion system using performance measures such as recall and precision. The graphical result shows that Norm_CombMNZ provides fused resulting text documents to the user, in the form of effective text retrieval.

Keywords
Data fusion, information retrieval, performance measures, IR models.

1. INTRODUCTION
A typical IR system allows users to express an information need in the form of a query, which is used to identify documents that contain information relevant to the user's request, thus satisfying the need. But no single approach to IR has been shown to achieve superior performance in all situations. This is because of the different methods of representing documents and user queries, the different policies regarding document and query pre-processing and the different algorithms used to rank documents. As a result, individual IR systems will retrieve different documents from the same document collection in response to the same query. The retrieval performance can be improved by combining the lists of documents produced by a number of different IR algorithms into a single list. This is known as data fusion [1].

Metasearch engines [2] are considered as an application of fusion to document retrieval; where a query is sent to a number of traditional search engines, each search engine returns a ranked list, metasearch engine fuse them to produce a single ranked list, which is better than any individual returned ranked lists.

This paper is organized as follows: Section II describes the problem that data fusion is intended to solve. Section III outlines previous work that has been undertaken in this field. In section IV, individual IR model is described. An algorithm for data fusion, Norm_CombMNZ is outlined in Section V. Section VI describes the experiments conducted to evaluate the performance of the Norm_CombMNZ algorithm with individual IR system. Finally, conclusion is outlined in section VII.

2. PROBLEM DESCRIPTION
Individual IR systems [3] based on Vector space, Smart Vector space, Extended Boolean, Latent semantic Indexing, etc. retrieve different documents in response to the same queries when operating on the same document collection. This may be as a result of difference in policies regarding query or document preprocessing, the algorithms used and representations of documents and queries.

Retrieval performance has been shown to be improved by fusing the result sets produced by a number of different IR systems into a single result set. Combining evidence from different systems leads to performance improvement. A number of different approaches to data fusion are outlined in section III. Vogt and Cottrell [4] identify three "effects", any of which can be leveraged by a fusion technique.

1) Chorus effect; this effect occurs when several retrieval approaches suggest that an item is relevant to a query; this tends to be stronger evidence for relevance than that of a single approach.

2) Skimming effect; this effect suggests that the relevant documents are most likely to occur on the top of the retrieved list for each individual retrieval system, so the fusion algorithm who skims the top ranked documents from each individual retrieval system is expected to be more efficient.

3) Dark horse effect; it assumes that the good fusion algorithm should treat the systems which retrieve larger number of relevant documents differently than other systems which don't retrieve large number of relevant documents.

The data fusion system presented here exploits Chorus, Skimming and Dark-Horse effects of data fusion.

3. BACKGROUND RESEARCH
Data fusion algorithms take N input lists from N different retrieval systems to fuse them, and as an output, it computes a single ranked list, which is an improvement over any input list as measured by standard IR performance metrics.

Several data fusion methods [1] are developed which are classified based on whether they rely on rank or score, or whether they require training data or not. Score-based methods make use of the relevance score assigned to each document by each input system to calculate a score by which the fused result set will be ordered. Relevance scores may not be available from all types of input system. For this reason, some rank-based have been developed that take result sets in the form of ranked lists of documents without scores as their input. These are fused using the position of each document in each result set.

Many approaches that use score or rank based methods have been proposed. Some of the score-based techniques are as follows:-

Bartell et al. [5] investigated the linear combination method. Numerical optimization techniques were used to determine optimal weights for component systems and positive results were achieved. Usually when using this method, some training queries and evaluations are required to determine the performance of those systems involved. This work demands a lot of human effort.

Fox and his colleagues [6] introduced a group of data fusion methods including CombSum and CombMNZ.

CombSum sets the score of each document in the combination to the sum of the scores obtained by the component results, while in CombMNZ the score of each document is obtained by multiplying this sum by the number of results which have non-zero scores.

Other Comb* algorithms include:-CombMIN- takes the minimum of individual relevance scores.

CombMED- takes the median of individual relevance scores.

CombMAX – takes the maximum of individual relevance scores.

Scores are not always available. For example, very few web search engines provide scores for the retrieved web documents. Suppose that only ranking information was available, Montague and Aslam suggested a few rank-based methods including Borda count [2] and Condorcet fusion [7].

Borda count method - The highest ranked document in a system gets n Borda points and each subsequent gets one point less where n is the number of total retrieved documents by all systems.

Condorcet method - The winner is the document which beats each of the other documents in a pair wise comparison.

D. Lillis proposed a probabilistic approach for data fusion called, ProbFuse [8] which uses the probability that particular documents are relevant to a given query.

4. INFORMATION RETRIEVAL MODELS

Numerous IR models [3] have been proposed to solve the problem of identifying documents in a collection that are relevant to a given query. These IR techniques typically assign a score to each document in a collection. This score is a judgment of that document's relevance to the given query. A list of documents, ranked according to this relevance score, is then returned. The Vector Space model, Extended Boolean model and Latent Semantic Indexing are used to produce inputs to this data fusion system.

4.1 Boolean Model

The Boolean Model [3] of Information Retrieval is based on Boolean algebra and set theory. Under this model, a document in the collection is judged to be relevant or non-relevant to the given query by considering whether query terms appear in the document or not.

Its principal advantage lies in its simplicity, in that for each relevance judgment, no information other than the query and the document itself is required. Its query language is also simple, consisting of keywords linked with the operators AND, OR and NOT. Its main drawback is that it cannot distinguish between levels of relevancy and fails to return partial matches, since its judgment is a binary one.

4.2 Vector-Space Model

The Vector Space Model [3] relies on a non-binary weighting system to rank documents in order of their relevance to a given query. Each term is assigned a weight in respect of each document. Each document is then represented by a t-dimensional vector of these weights, where t is the total number of terms present in the document collection. The similarity between a query and a document is then calculated as the cosine of the angle between their respective vectors.

A key benefit of the Vector Space Model is that a document will not be automatically judged as being non-relevant as a result of not containing all the terms in the query. A similarity threshold may be introduced to prevent documents with too low a similarity score being returned by the system.

The most popular method for calculating the weights for each term relies on term frequency (tf) and inverse document frequency (idf). The tf represents how often the term appears in the document in question. The reasoning behind this is that if a query term occurs frequently in a document, it is more likely to be relevant to the query. Similarity is computed by comparing the deviation of angles between each document vector and the original query vector. i.e. by finding the cosine of angle between them. A cosine value of zero means that the query and document vector are orthogonal and have no match.

4.3 Probabilistic Model

A Probabilistic IR system [3] attempts to calculate the probability of each document being relevant to a particular query. As with the Vector Space Model, relevance is not a binary attribute, so some documents may be judged to be more relevant than others. For best performance, the model relies on information about relevant documents to estimate the relevance of other documents. In many systems (including a web-based search engine), no relevance information is available when the search is initiated. Thus tf and idf can be used to estimate relevance at this stage. Relevance information is frequently obtained by using user feedback to refine the search.

4.4 Extended Boolean Model

The Extended Boolean Model [3] extends the simple Boolean Model to allow partial matching and term weighting by incorporating elements of the Vector Space Model, namely tf and idf.

The weights used in the Extended Boolean Model are slightly different to those used in the Vector Space Model. Instead of using the t f-idf weighting scheme in vector space, a normalized variation is used, which ensures that the weights lie between 0 and 1.

4.5 Latent Semantic Indexing

Latent semantic indexing [3] adds an important step to the document indexing process. In addition to recording which keywords a document contains, the method examines the document collection as a whole, to see which other documents contain some of those same words.

LSI considers documents that have many words in common to be semantically close, and ones with few words in common to be semantically distant. This simple method correlates well with how a human being, looking at content, might classify a document collection. Although the LSI algorithm doesn't understand anything about what the words mean, the patterns it notices can make it seem astonishingly intelligent.

LSI maps each document and query vector into a lower dimensional space which is associated with concepts. This is accomplished by mapping the index terms vectors into this lower dimensional space. The LSI proposes to decompose a term-document association matrix in three components using singular value decomposition. The first one is the matrix of eigenvectors derived from the term-to term correlation matrix; the second one is the matrix of eigenvectors derived from the transpose of the document-to document matrix; the third one is a r×r diagonal matrix of singular values where r is the minimum between the row and the column of the original matrix, and the rank of the term-document association matrix.

5. DATA FUSION ALGORITHM

CombMNZ adds all the reported scores for a document and multiplies the sum value to the number of retrieval models that have returned that document (d).

$$CombMNZ_d = \sum_{c}^{N} D^c \times |D^c > 0| \quad (1)$$

D^c is the normalized score of document d in result c.

$|D^c > 0|$ is the no. of non-zero normalized scores given to d by any result set.

N is the number of result sets to be fused.

$$D^c = \frac{S^d - D^c{}_{min}}{D^c{}_{max} - D^c{}_{min}} \quad (2)$$

S^d is the score of document d in the rank list c before normalization.

D^cmin and D^cmax are the minimum and maximum document.

Score normalization is required when final result of CombMNZ is obtained, as specified in equation 2. This added normalization step results in scores to lie within the common range of 0 to 1. So, the documents with score 1 can be considered as most relevant, those with scores above 0 and less than 1 as partially relevant and documents with score 0 as non-relevant.

$$\text{NormCombMNZ}_d = \frac{\text{CombMNZ}_d - \text{CombMNZ}_{min}}{\text{CombMNZ}_{max} - \text{CombMNZ}_{min}} \quad (3)$$

$NormCombMNZ_d$ is Normalized score obtained by NormCombMNZ for document d.

$CombMNZ_d$ is the CombMNZ score for document d.

$CombMNZ_{min}$ is minimum score from CombMNZ scores.

$CombMNZ_{max}$ is maximum score from CombMNZ scores.

Table 1. Norm_combmnz example

Doc no.	IR model1 score	IR model2 score	Normalized IR Model1 score(Dc1)	Normalized IR Model2 score(Dc2)	Sum(Dc)
1	0.0059175	0.0869288	0.01962541	0.416702287	0.436328
2	0.0024715	0.1658677	0.018924528	0.795104424	0.814029
3	0.0057061	0	0.018924528	0	0.018925
4	0.0032976	0	0.010936543	0	0.010937
5	0	0	0	0	0
6	0.3015203	0.2086112	1	1	2
7	0.0023323	0.0741351	0.007735133	0.355374465	0.36311
8	0.0032314	0	0.010716889	0	0.010717
9	0.2244579	0.1507141	0.74442055	0.722464169	1.466885

Table 1. Norm_combmnz example(cont..)

Doc no.	sum(Dc)*nonzero scores from IR model1 (i.e. 8)	Norm_CombMNZ Score	sum(Dc)*nonzero scores from IR model2(i.e.5)	Norm_CombMNZ Score	Rank
1	17.4531079	0.21816385	2.181638484	0.21816385	4
2	32.5611581	0.40701448	4.07014476	0.40701448	3
3	0.75698113	0.00946226	0.094622641	0.00946226	6
4	0.43746171	0.00546827	0.054682714	0.00546827	7
5	0	0	0	0	9
6	80	1	10	1	1
7	14.5243839	0.1815548	1.815547991	0.1815548	5
8	0.42867557	0.00535844	0.053584447	0.00535844	8
9	58.6753888	0.73344236	7.334423596	0.73344236	2

6. EXPERIMENTS AND PERFORMANCE EVALUATION

The performance of a data fusion system can be measured with respect to other information retrieval systems using measures such as precision and recall. Precision and Recall [3] are evaluation measures that reflect the key aims of any IR system.

Recall measures a system's success at returning the maximum number of relevant documents possible. Precision measures the ability to avoid returning non-relevant documents. Recall is the fraction of the total available relevant documents that have been retrieved. It is given by

$$\text{Recall} = |Ra|/|R| \quad (4)$$

where |Ra| is the number of relevant documents that have been retrieved and |R| is the total number of relevant documents that are contained in the document collection.

Precision is the fraction of the total number of documents that have been retrieved that are relevant. It is given by

$$\text{Precision} = |Ra|/|A| \quad (5)$$

where |Ra| is the number of relevant documents that have been retrieved and |A| is the number of documents in the result set that is returned.

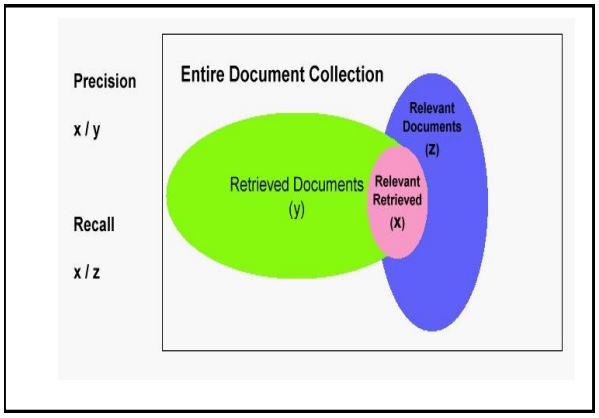

Fig 1: Precision and Recall

A corpus is maintained on local machine containing text documents on sport cricket news articles. Different queries are sent to the data fusion system and performance evaluation is carried out on result.

For example:-

Query: spot fixing scandal

Relevant documents: 25

{d1,d2,d6,d7,d8,d12,d13,d14,d20,d21,d27,d45,d46,d47,d48, d49,d51,d55,d56,d59,d60, d69, d70,d89,d92,d93 }

Vector Space model retrieved documents: 24

{d49,d51,d70,d60,d75,d67,d11,d94,d46,d1,d89,d97,d13,d48, d78,d12,d72,d52,d14,d45, d22, d8,d92,d47 }

Smart Vector Space model retrieved documents: 31

{d49,d51,d75,d68,d67,d11,d94,d70,d27,d46,d1,d60,d48,d13 ,d39,d88,d24,d71,d16,d12,d72,d52,d78,d8,d22,d45,d14,d97, d47,d92 }

Extended Boolean model retrieved documents: 31

{d49,d51,d68,d94,d60,d67,d1,d70,d27,d89,d46,d39,d48,d71, d11,d75,d13,d8,d12,d22,d45,d16,d24,d88,d14,d92,d78,d72, d52, d97,d47 }

Latent Semantic Indexing model retrieved documents: 41

{d69,d92,d45,d8,d10,d55,d59,d20,d6,d61,d57,d27,d47,d60,d2 1,d89,d70,d48,d53,d54,d46,d13,d49,d1,d56,d51,d41,d14,d62, d26,d82,d7,d83,d42,d2,d58,d52,d38,d93,d12,d86}

Norm_CombMNZ retrieved documents: 55

{d49,d51,d70,d60,d46,d1,d89,d48,d14,d27,d8,d45,d94,d67,d6 8,d75,d92,d14,d47,d11,d12,d69,d10,d55,d59,d20,d6,d61,d57, d21,d52,d53,d54,d56,d42,d62,d26,d82,d7,d83,d42,d2,d58,d97 , d39,d38,d93,d22,d86,d71,d72,d88,d24,d16,d78 }

Precision and Recall is calculated at different levels are shown in Table 2:-

Table 2. Precision & Recall of Individual IR System v/s Norm_CombMNZ

Recall level	VS Precision	SVS Precision	EXTB Precision	LSI Precision	Norm_CombMNZ Precision
0	100	100	100	100	100
10	100	100	100	100	100
20	55.55	50	62.5	83.33	100
30	63.63	58.33	70	87.5	100
40	62.5	66.66	58.82	76.9	100
50	60	48	63.16	80	100
60	62.5	50	57.7	83.33	78.95
70	0	51.61	51.61	77.27	77.3
80	0	0	0	80	76.92
90	0	0	0	78.57	73.33
100	0	0	0	64.1	64.1

The following graphical result is seen:-

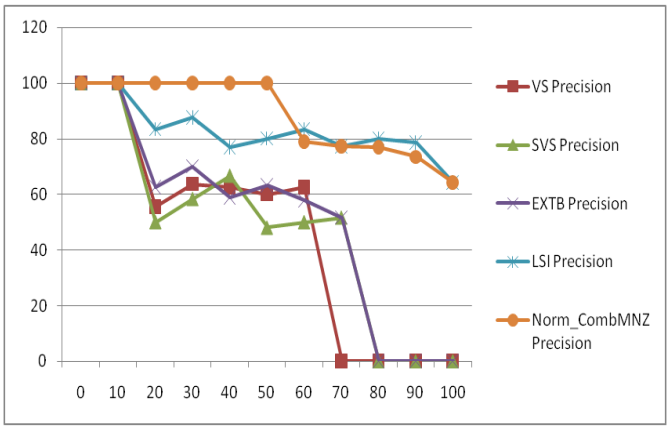

Fig 2: Graphical result for given Query

The graphical result of both queries indicates that Norm_CombMNZ data fusion gives comparatively better result than other IR models.

LSI model almost matches the performance of Norm_CombMNZ.

The no. of documents retrieved by fusion is more than other IR systems.

The relevant documents appear on top of retrieved list, which exploits skimming effect of fusion.

All IR systems retrieve different number of relevant documents, which exploits dark horse effect of fusion.

Certain documents are retrieved by more than one IR system, which are more relevant and are also retrieved by fusion system. This exploits chorus effect of fusion.

As compared to CombMNZ, a Norm_CombMNZ provides better ranking judgement since scores lie within a common range of 0 to1.

7. CONCLUSION

A data fusion system combines the document similarity scores from different information retrieval models. This paper focuses on system that provides a more effective result to user as compared to individual IR systems, which retrieve different documents when same document collection is used. The effectiveness of a system is evaluated using the performance measures and with graphical results.

A better ranking judgement is also provided by normalizing the fused scores in a range from 0 to1. This Norm_CombMNZ data fusion system exploits Chorus, Skimming and Dark-Horse effects of data fusion.

8. REFERENCES

[1] Mohammad Othman Nassar, Ghassan Kanaan, "The Factors Affecting the Performance of Data Fusion Algorithms," icime, pp.465-470, 2009 International Conference on Information Management and Engineering, published by IEEE press, 2009, ISBN:978-0-7695-3595-1.

[2] Javed Aslam, and Mark Montague, "Models for Metasearch," In Proc. ACM SIGIR 2001 Conf., ACM press, New Orleans, Louisiana, 2001, pp. 276-284.

[3] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. Modern Information Retrieval. Addison-Wesley Longman Publishing Co., Inc., Boston,USA, 1999.

[4] Christopher Vogt and Garrison Cottrell,"Fusion via a linear combination of scores," Information Retrieval, 1(3), Oct. 1999, pp.151-173.

[5] Bartell, B. T., Cottrell, G. W., & Belew, R. K., "Automatic combination of multiple ranked retrieval systems" in Proceedings of ACM SIGIR conference (p.173-184), 1994, Dublin, Ireland.

[6] Belkin, N. J., Kantor, P., Fox, E. A., & Shaw, J. A. "Combining evidence of multiple query representations for information retrieval". Information Processing & Management, 1995, 31 (3): 431-448.

[7] Montague, M., & Aslam, J. A., "Condorcet fusion for improved retrieval", in Proceedings of ACM CIKM conference (p. 538-548), 2002, McLean, VA, USA.

[8] David Lillis, Fergus Toolan, Rem Collier, and John Dunnion, "ProbFuse: a probabilistic approach to Data-fusion," in Proc. 29th ACM SIGIR conf., ACM press, Seattle, Washington, USA, 2006, pp.139-146.

[9] Hai Dong, Farookh Khadeer Hussain,Elizabeth Chang "A Survey in Traditional Information Retrieval Models", 2008 Second IEEE International Conference on Digital Ecosystems and Technologies.

[10] Shengli Wu, "Applying statistical principles to data fusion in information retrieval," Expert Systems with Applications: An International Journal, Volume 36 , Issue 2, March 2009,Pergamon Press, pp. 2997-3006.

[11] Shengli Wu, Fabio Crestani, and Yaxin Bi, "Evaluating score normalization methods in Data-fusion," Springer Berlin, 2006, pp. 642–648.

[12] D. Frank Hsu and Isak Taksa"Comparing Rank and Score Combination Methods for Data Fusion in Information Retrieval", 2005 Springer Science + Business Media, Inc.

[13] Shengli Wu and Sally McClean, "Performance prediction of data fusion for information retrieval," Information Processing and Management, Vol. 42, Issue 4, Elsevier, 2006, pp. 899-915.

[14] Beitzel, S., Jensen, E., Chowdhury, A., Grossman,D., Frieder, O., & Goharian, N. "On fusion of effective retrieval strategies in the same information retrieval system", Journal of the American Society of Information Science and Technology, 2004: 55 (10), 859-868.

[15] Martin F. Porter. An algorithm for suffix stripping. Pages 313–316, 1997.