A Geometric Probabilistic Framework for Data Fusion in Information Retrieval

Shengli Wu
School of Computing and Mathematics
University of Ulster
Newtownabbey, Northern Ireland, UK.
s.wu1@ulster.ac.uk

Abstract - Data fusion in information retrieval has been investigated by many researchers and quite a few data fusion methods have been proposed, but why data fusion can bring improvement in effectiveness is still not very clear. In this paper, we use a geometric probabilistic framework to formally describe data fusion, in which each component result returned from an information retrieval system for a given query is represented as a point in a multiple dimensional space. Then all the component results and data fusion results can be explained using geometrical principles. In such a framework, it becomes clear why quite often data fusion can bring improvement in effectiveness and accordingly what the favourable conditions are for data fusion algorithms to achieve better results. The framework can be used as a guideline to make data fusion techniques be used more effectively.

Keywords: Information retrieval, data fusion, meta-search, geometric probabilistic framework, evaluation.

1 Introduction

The goal of information retrieval systems is to provide useful information for the user’s information need. For a collection of documents (such as a digital library or the web) and a given query, an information retrieval system returns a ranked list of documents. Binary relevance judgment is very often used in retrieval evaluation. It divides all documents into two categories: relevant and irrelevant. Precision (number of relevant documents retrieved/total number of documents retrieved) and recall (number of relevant documents retrieved/total number of relevant documents in the whole collection) and their derivations are commonly used to evaluate the effectiveness of information retrieval results.

In information retrieval, a variety of representation techniques for queries and documents have been proposed, and many retrieval techniques have also been developed for obtaining higher retrieval effectiveness. These techniques are comparable in performance and there is no all-time winner. In such a situation, using a few independent information retrieval systems or using one single retrieval system but several different query representations or several different parameter settings to search the same document collection for the same information need, and then fusing these results from different information retrieval systems for better retrieval effectiveness is an attractive option. This is the primary idea behind data fusion.

Data fusion (also known as meta-search) in information retrieval has been investigated by many researchers and quite a few data fusion algorithms such as CombSum [9, 10], CombMNZ [9, 10], linear combination [3, 24, 25], Borda fusion [2], Condorcet fusion [17], the correlation methods [28] have been proposed and extensive experimentation has been conducted to evaluate these algorithms. Among these data fusion algorithms, CombSum is the most important one because many other data fusion algorithms (all these listed above except Condorcet fusion) can be regarded as variations of it.

Condorcet fusion is very different from the others. It is borrowed from the political science idea of majority voting. It considers all possible head-to-head ranking competitions among all possible document pairs. Then these documents can be ranked according to the number of competitions they have won. However, Condorcet fusion’s high time complexity is a problem for application. In this paper we do not investigate data fusion methods like Condorcet fusion. We focus our attention on data fusion methods such as CombSum and CombMNZ, which use a function to calculate scores for all the documents, and then rank them according to the calculated scores.

A key question about data fusion is: why can data fusion bring improvement in effectiveness? One answer to this question can be “the multiple evidence principle” [14]. This is generally true because: the more times a particular document appears in component results, the more likely it is relevant to the information need. However, this is not very precise. Each of the proposed data fusion methods can be regarded as using this “multiple evidence principle” in a particular way, but the multiple evidence principle does not tell us which one is the optimum. For example, let us consider CombSum and CombMNZ. Extensive experimentation (e.g., in [2, 9, 10, 14, 16, 27, 28]) has been conducted to evaluate them with TREC data. Their performances are

---

1 TREC (Text REtrieval Conference) is an annual information retrieval evaluation event held by the National Institute of Standards and Technology and the Advanced Research and Development Activity (ARDA)
always close. For some data collections, CombMNZ is better than CombSum; while for some other collections, CombSum is better than CombMNZ. It seems that they are very close. It is interesting to find out why this is the case and what the favourable conditions are for them.

In this paper we would like to investigate these issues through a formal geometrical probabilistic framework. In this framework, all results (either from component information retrieval systems or from data fusion algorithms) are represented as points in a multi-dimensional space. As a result, it becomes clear which conditions are favourable for data fusion and which algorithm (e.g., CombSum or CombMNZ or some other methods) is the optimum in certain conditions. The outcomes of this paper can be used to improve the effectiveness of existing data fusion techniques.

The rest of this paper is organized as follows: Section 2 introduces the framework of data fusion based on geometrical principles. In this framework, every component result as well as the fused result is represented as a point in a multi-dimensional space. Then in Sections 3 we discuss some further characteristics of data fusion using this framework. Section 4 presents the relationship between Euclidean distance and some other commonly used measures in information retrieval. In Section 5 we review some related work on data fusion in information retrieval. Section 6 is the conclusion.

2 A geometric probabilistic framework for data fusion

Suppose we have a collection of documents which includes n documents $D=(d_1, d_2, ..., d_n)$ and a group of information retrieval systems $R=(r_1, r_2, ..., r_p)$. For a given query $q$, an information retrieval system $r_i$ returns a set of scores $S_i=(s_{i1}, s_{i2}, ..., s_{in})$ as the result. Here each $s_i$ denotes the probability score of relevance of document $d_i$ to $q$. For any $s_i$ its range is between 0 and 1 inclusive. If we only consider these results on their own, then each of them can return a ranked list of documents whose scores decreases with rank. In case of data fusion, we need to consider all these component results to determine the best possible ranking for all the documents involved. The geometric probabilistic framework is aimed to provide a formal framework for such a purpose.

Let us consider a n-dimensional space, in which the relevance probability of each document is regarded as a dimension, then any retrieval result $S=(s_1, s_2, ..., s_n)$ must be located inside the hypercube $(p_n;0-1, p_n;0-1, ..., p_n;0-1)$, which is referred to as the result space for a given query. It is denoted as $X(q, D)$, where $q$ is the query and $D$ is the document collection. For any query, the number of dimensions in its result space is the number of documents in $D$.

For the result space $X(q, D)$ of a given query $q$ and a given document collection $D$, one result $O=(o_1, o_2, ..., o_n)$ is the optimum point, if every $o_i$ is the real relevance probability score of the corresponding document. For any result $S=(s_1, s_2, ..., s_n)$, its effectiveness can be evaluated by calculating the Euclidean distance (dis) of this result and the optimum point in the result space (i.e., $dis(S, O)=\sqrt{\sum_{i=1}^{n}(s_i-o_i)^2}$). Its relationship with some other measures will be discussed in Section 4. Usually, there are a group of results which have an equal distance to the optimum point (equally effective).

In a result space, data fusion can be described as finding a result (point) $F$ from a group of results (points) $S=(s_1, s_2, ..., s_n)$ of a given query $q$. We expect that $F$ can be as close to the optimum point as possible.

For a group of results (points) $S=(s_1, s_2, ..., s_n)$, there is a point $C$ which is the centroid of these points $S_1, S_2, ..., S_n$. Set $C=(c_1, c_2, ..., c_n)$, then

$$c_i = \frac{1}{m} \sum_{j=1}^{m} s_{ij}$$

for any $i (1 \leq i \leq n)$. Using the centroid can be regarded as the best way of merging these points, when no other information about these results is available. The following Theorem 1 proves this.

**Theorem 1.** In a n-dimensional space $X$, there are $m$ points $S=(s_1, s_2, ..., s_n)$ (1≤j≤m), and $C=(c_1, c_2, ..., c_n)$ is the centroid of $S_j (1 \leq j \leq m)$. Then $C$ is the point which has the shortest average distance to $S_1, S_2, ..., S_n$.

**Proof:** Given $m$ points $S=(s_1, s_2, ..., s_n)$ (1≤j≤m), we are going to find a point $P=(p_1, p_2, ..., p_n)$ in $X$ such that the average distance between $P$ to $S_j (1 \leq j \leq m)$ is the shortest. Now let $F$ be the point such that the sum of squares of distances between $P$ to $S_j (1 \leq j \leq m)$ is the shortest. Let $F$ be the sum of squares of distances between $P$ to $S_j (1 \leq j \leq m)$. Then

$$F = \sum_{j=1}^{m} (P - S_j)(P - S_j)^T = \sum_{j=1}^{m} \sum_{i=1}^{n} (p_i - s_{ij})^2$$

$P$ is the optimum solution if and only if

$$\frac{\partial F}{\partial p_i} = 2 \sum_{j=1}^{m} (p_i - s_{ij}) = 0, i=1, ..., n$$

$$\Rightarrow p_i = \frac{1}{m} \sum_{j=1}^{m} s_{ij}, i=1, ..., n$$

Therefore, the centroid of $S_1, S_2, ..., S_n$ is the point we are looking for and this completes the proof. •

**Theorem 2.** In a n-dimensional space $X$, there are $m$ points $S=(s_1, s_2, ..., s_n)$ (1≤j≤m), and $C=(c_1, c_2, ..., c_n)$ is the centroid of $S_1, S_2, ..., S_n$. For any target point $P=(p_1, p_2, ..., p_n)$ in $X$, the distance between $P$ and $C$ is no greater than the average distance between $P$ and $m$ points $S_1, S_2, ..., S_n$, i.e.,
The “Minkowski Sum Inequality” [29] can be used to prove this theorem. However, due to space limit, proof of this theorem and some other theorems later in this paper is omitted.

**Corollary 2.1.** Theorem 2 holds if \( P \) is the optimum point \( O \).

Theorem 2 and Corollary 2.1 tell us that the fused result by centroid-based method is at least as effective as the average performance of all component results. This can be regarded as a good property of the centroid-based data fusion method.

**Corollary 2.2.** In a \( n \)-dimensional space \( X \), there is a group of points \( S= (S_1, S_2, ..., S_m) \) and \( C \) is the centroid of \( S \). Suppose that \( O \) is the optimum point and every \( S_j \) (1\(\leq m\)) has equal distance to \( O \). Then the distance between \( C \) and \( O \) is no greater than the distance between each \( S_j (j=1, ..., m) \) and \( O \).

The above corollary specifies that if all component results are equally effective, then the fused result by calculating their centroid is at least as effective as any of the component results. Therefore, this is a favourable condition for data fusion and we should make it happen.

**Theorem 3.** In a \( n \)-dimensional space \( X \), suppose a group of \( m \) points \( S= (S_1, S_2, ..., S_m) \) and \( C \) is the centroid of \( S \). Then the distance between \( C \) and \( O \) is the average distance between \( O \) and \( S_1, S_2, ..., S_m \).

This theorem demonstrates that the minimal limit of effectiveness is achievable for centroid-based fusion. This is the worst situation for data fusion. Therefore, we should avoid this situation from happening.

**Example 1.** A collection includes 4 documents \( D=\{d_1, d_2, d_3, d_4\} \), and 3 information retrieval systems \( IR=\{ir_1, ir_2, ir_3\} \) are used. For a given query \( q \), an information retrieval system \( ir_j \) (j=1, 2, 3) returns a set of scores \( S=\{s_{1j}, s_{2j}, s_{3j}, s_{4j}\} \). Then, the distance between any \( S_j \) and \( S \) is:

\[
d_{ij} = \sqrt{(0.8 - 0.7)^2 + (0.6 - 0.5)^2 + (0.4 - 0.3)^2 + (0.2 - 0.1)^2}
\]

\= 0.200

\[
d_{i1} = \sqrt{(0.6 - 0.7)^2 + (0.6 - 0.5)^2 + (0.5 - 0.3)^2 + (0.3 - 0.1)^2}
\]

\= 0.333

**Theorem 4.** Suppose that \( S_1, S_2, ..., S_m \) and the optimum point \( O \) are known points. The distance between \( S_i \) and \( S \) is denoted as \( d_{iS} \), the distance between any \( S_i \) and \( O \) is denoted as \( d_{iO} \) (j=1, ..., m), and the centroid of \( S_1, S_2, ..., S_m \) is \( C \), then the distance between \( C \) and \( O \) is:

\[
d_{CO} = \frac{1}{m} \sum_{j=1}^{m} d_{jO} - \frac{1}{m} \sum_{k=1}^{m-1} d_{kS_k} \]

Note that shorter distances represent better results. We have verified that \( \text{ave} > d_{CO} \) in this case. Therefore, the fused result (centroid \( C \)) is better than the average of component results, and \( C \) (0.160) is better than the best of all three component results (\( S_1 \); 0.200).

**3 Position of component results**

**Theorem 4.** Suppose that \( S_1, S_2, ..., S_m \) and the optimum point \( O \) are known points. The distance between \( S_i \) and \( O \) is denoted as \( d_{iO} \), the distance between any \( S_i \) and \( O \) is denoted as \( d_{iO} \) (j=1, ..., m), and the centroid of \( S_1, S_2, ..., S_m \) is \( C \), then the distance between \( C \) and \( O \) is:

\[
d_{CO} = \frac{1}{m} \left( \sum_{j=1}^{m} d_{jO} - \sum_{k=1}^{m-1} d_{kS_k} \right)
\]

Theorem 4 tells us that \( d_{CO} \) can be represented by \( d_{iS} \) and \( d_{iO} \) for \( 1 \leq k \leq m \) and \( 1 \leq i \leq m \). To make \( d_{CO} \) as short as possible, it requires that \( d_{CO} \) should be as short as possible and \( d_{iS} \) should be as long as possible. That is, all component results should be as effective as possible, and all pairs of component results should be as different as possible.

**Example 2.** Let \( m=2 \), \( d_{1S_1} = 12 \), \( d_{2S_1} = 10 \), \( d_{1S_2} = 8 \).

\[
d_{CO} = \frac{1}{2} \left[ 2 \cdot \sqrt{d_{1O}^2} + 2 \cdot \sqrt{d_{2O}^2} - d_{1S_1}^2 \right]
\]

\= 10.30

The fused result is more effective than \( S_1 \) since 10.30<12, and less effective than \( S_1 \) since 10.30>10.

**Corollary 4.1.** There are two groups of points: \( S_1=\{ S_{11}, S_{12}, ..., S_{1n}\} \) and \( S_2=\{ S_{21}, S_{22}, ..., S_{2n}\} \). \( C_1 \) is the centroid of \( S_1, S_{21}, ..., S_{2n} \) and \( C_2 \) is the centroid of \( S_1, S_{21}, ..., S_{2n} \). 

\[
\text{ave} = \sqrt{(0.9 - 0.7)^2 + (0.4 - 0.5)^2 + (0.3 - 0.3)^2 + (0.1 - 0.1)^2}
\]

\= 0.224

Then fusing these results by calculating their centroid, we obtain:

\[
c_1 = \frac{1}{3} \sum_{j=1}^{3} s_{ij} = \frac{1}{3} (0.8 + 0.6 + 0.9) = 0.767
\]

\[
c_2 = \frac{1}{3} \sum_{j=1}^{3} s_{2j} = \frac{1}{3} (0.6 + 0.6 + 0.4) = 0.533
\]

\[
c_3 = \frac{1}{3} \sum_{j=1}^{3} s_{3j} = \frac{1}{3} (0.4 + 0.5 + 0.3) = 0.400
\]

\[
c_4 = \frac{1}{3} \sum_{j=1}^{3} s_{4j} = \frac{1}{3} (0.2 + 0.3 + 0.1) = 0.200
\]

Its effectiveness is:

\[
d_{CO} = \sqrt{(0.767 - 0.7)^2 + (0.533 - 0.5)^2 + (0.4 - 0.3)^2 + (0.2 - 0.1)^2}
\]

\= 0.160

The average effectiveness of the three component results is:

\[
\text{ave} = (0.2000 + 0.3333 + 0.2236)/3 = 0.253
\]

Note that shorter distances represent better results. We have verified that \( \text{ave} > d_{CO} \) in this case. Therefore, the fused result (centroid \( C \)) is better than the average of component results, and \( C \) (0.160) is better than the best of all three component results (\( S_1 \); 0.200).
Theorem 5 can be true in more general situations in distance between the centroids of points and the optimum point is shorter than the average to say, the distance between the centroid of a group of the same. That in general better performance can be achieved if we know points. The centroid of all component results affects data fusion. This can be ascertained from Equation 3.

And the following three conditions hold:
1. \( \sum_{k=1}^{m-1} \sum_{j=k+1}^{m} \text{dis}(S_{k1}, S_{1j}) = \sum_{k=1}^{m-1} \sum_{j=k+1}^{m} \text{dis}(S_{2k}, S_{2j}) \)
2. \( \text{dis}(C_{1}, O) < \frac{1}{m} \sum_{i=1}^{m} \text{dis}(C_{i}, O) \)
3. \( v_1 < v_2 \)

Then we have \( \text{dis}(C_{1}, O) < \text{dis}(C_{2}, O) \).

Corollary 4.1 tells us that the variance of effectiveness of all component results affects data fusion when other conditions are the same. The smaller the variance is, the better result we can obtain from data fusion. On the other hand, not like effectiveness, only the total dissimilarity does not affect the effectiveness of the result; however, some measures may not be sensitive enough to reflect all such changes. Anyhow, by any measure, the effectiveness of a result \( R \) should be at least as effective as \( R' \), if \( R' \) includes the same group of documents as \( R \) and all the documents in \( R' \) are in the same rank as in \( R \) except one relevant document and its subsequent irrelevant document in \( R \) reverse their ranks in \( R' \).

4. If one irrelevant document is followed by a relevant document, then exchanging their positions upgrades the effectiveness of the result. However, some measures may not be sensitive enough to reflect all such changes. Anyhow, by any measure, the effectiveness of a result \( R' \) should be at least as effective as \( R \), if \( R' \) includes the same group of documents as \( R \) and all the documents in \( R' \) are in the same rank as in \( R \) except one irrelevant document and its subsequent relevant document in \( R \) reverse their ranks in \( R' \).

For example, suppose we have two result: \( R_1=(r, i, i, r, i) \) and \( R_2=(i, r, i, r, i) \). Here ‘r’ denotes a relevant document and ‘i’ denotes an irrelevant document. \( R_1 \) and \( R_2 \) are different in rank 1 and rank 2. If we use \( AP \) (average precision over all relevant documents), then \( AP(R_1)=(1/1+2/4)/2=0.75 \) and \( AP(R_2)=(1/2+2/4)/2=0.5 \). \( R_1 \) is assigned a higher value than \( R_2 \). If we use \( P_5 \) (precision at first 5 cut-off documents), then \( P_5(R_1)=P_5(R_2)=2/5=0.4 \). The difference in \( R_1 \) and \( R_2 \) does affect \( AP \) but not \( P_5 \). This example illustrates that \( P_5 \) is not as sensitive as \( AP \), though they are reasonable measures since the four rules above are observed by both of them.

Theorem 6. Using the geometric probabilistic framework and any measure which observes the four rules above, the statistically best way of ranking a
group of documents is to rank them according to their probabilities of relevance from the highest to the lowest.

One thing we should note is that the statistically best way of ranking in the sense of probability of relevance does not guarantee that the ranking generated in such a way is the best in every single case when binary relevance judgment is applied, but they are the best in the long run if enough cases have been considered.

Euclidean distance is a measure which considers all the documents in the result. Suppose \( S = (s_1, s_2, \ldots, s_n) \) and \( O = (o_1, o_2, \ldots, o_n) \), then \( \text{dis}(S, O) = \sqrt{\sum_{i=1}^{n} (s_i - o_i)^2} \). For every element \( s_i \) in \( S \), we can define its distance from the corresponding element \( o_i \): \( \text{dis}(s_i, o_i) = \sqrt{(s_i - o_i)^2} \). Here \( \text{dis}(s_i, o_i) \) is referred to as the distance of the \( i \)-th element between every point in \( G_1 \) and \( O \).

**Theorem 7.** \( O = (o_1, o_2, \ldots, o_n) \) is the optimum point in an \( n \)-dimensional space \( X \). Here each \( o_i \) is the accurate relevance probability value of document \( d_i \) (\( 1 \leq i \leq n \)). \( G_1 \) is a group of points, and each of them has an equal distance of \( a_i \) from \( O \). \( G_2 \) is another group of points in \( X \), and each of them has an equal distance of \( a_i \) from \( O \). \( a_i \leq a_{i+1} \). Then for any dimension \( i \), the average distance of the \( i \)-th element between every point in \( G_1 \) and \( O \) is shorter than the average distance of the \( i \)-th element between every point in \( G_2 \) and \( O \).

For a group of documents, we can rank them in different ways. Suppose that \( D_1 \) and \( D_2 \) are two different ranked lists of the same group of documents, we can always transform \( D_1 \) to \( D_2 \) and vice versa by repeated neighbour swaps. For example, if \( D_1 = (d_1, d_2, d_3) \) and \( D_2 = (d_2, d_1, d_3) \) then we can transform \( D_1 \) to \( D_2 \) by swapping the positions of \( d_1 \) and \( d_2 \) and the positions of \( d_1 \) and \( d_3 \). In this case, we say that the two rankings \( D_1 \) and \( D_2 \) are two steps away: \( \text{swap\_dis}(D_1, D_2) = 2 \). The dissimilarity of any two rankings of documents can be evaluated by the number of neighbour swaps needed for transforming one ranking to the other.

For a group of documents \( D = (d_1, d_2, \ldots, d_n) \), an estimation of their probabilities of relevance forms a point \( S = (s_1, s_2, \ldots, s_n) \) in an \( n \)-dimensional space \( X \). All the documents in \( D \) can be ranked according to their relevance probability scores from the highest to the lowest. \( \text{Doc}(S) \) is used to denote the ranking we obtain from \( S \). If \( S' = (s'_1, s'_2, \ldots, s'_n) \) is another point in \( X \), then \( \text{Doc}(S') \) and \( \text{Doc}(S) \) are two different rankings of the same group of documents \( D = (d_1, d_2, \ldots, d_n) \).

**Theorem 8.** \( O = (o_1, o_2, \ldots, o_n) \) is the optimum point in an \( n \)-dimensional space \( X \). \( \text{Doc}(O) = (d_1, d_2, \ldots, d_n) \) is the ranking we obtain for \( O \). \( G_1 \) is a group of points, and each of them has an equal distance of \( a_i \) from \( O \). \( G_2 \) is another group of points in \( X \), and each of them has an equal distance of \( a_i \) from \( O \). \( a_i \leq a_{i+1} \). On average, transforming \( \text{Doc}(O) \) to the document ranking of the points in \( G_1 \) requires at least as more neighbour swaps as transforming \( \text{Doc}(O) \) to the document ranking of the points in \( G_2 \).

\( \text{Doc}(O) \) is the best document ranking for \( O \), where all the documents are ranked according to their actual relevance probability scores. For any other ranking \( \text{Doc}(S) \) of the same document collection, the more swap steps it needs for the transformation of \( \text{Doc}(S) \) to \( \text{Doc}(O) \), the less effective \( \text{Doc}(S) \) is. Theorem 8 proved that a positive relationship exists between \( \text{swap\_dis}(\text{Doc}(O), \text{Doc}(S)) \) and \( \text{dis}(O, S) \).

For a group of documents \( D = (d_1, d_2, \ldots, d_n) \), \( \text{Doc}(O) = (d_1, d_2, \ldots, d_n) \) is a ranked list of documents of \( O \). Their relevance probability scores are \( O = (o_1, o_2, \ldots, o_n) \). \( \text{Doc}(O) \) is optimum since \( o_1 \leq o_2 \leq \cdots \leq o_n \). \( S = (s_1, s_2, \ldots, s_n) \) is a point in \( X \). \( \text{Doc}(S) = (d_1, d_2, \ldots, d_n) \) is a ranked list of documents of \( S \). Let us discuss the relationship between swap distance and precision at first \( k \) cut-off documents \( (P_k) \). Precision at first \( k \) cut-off documents is a commonly used measure in retrieval evaluation.

If we swap \( d_i \) and \( d_j \) in \( \text{Doc}(O) = (d_1, d_2, \ldots, d_n) \), then the result is less effective this is because the relevance probability that the document at rank \( k \) decreases by \( s_{j_k} \) and the relevance probability that the document at rank \( i+1 \) increases by \( s_{i_k} \).

We divide the documents of \( \text{Doc}(O) = (d_1, d_2, \ldots, d_n) \) into two groups: the first \( k \) documents are in the first group, and the rest of the documents are in the second group. For the total of \( n \) documents, there are \( n*(n-1)/2 \) different swaps: \( k*(k-1)/2 \) different swaps for the first group and \( (n-k) *(n-k-1)/2 \) different swaps for the second group. Those swaps happen inside one group does not affect the value of \( P_k \). Therefore, among all \( n*(n-1)/2 \) different swaps, \( n*(n-1)/2-k*(k-1)/2-(n-k) *(n-k-1)/2 \) swaps will cause the value of \( P_k \) decreasing by \( 1/k \). We may define the ratio of swaps which affect \( P_k \) as

\[
\text{ratio} = \frac{n*(n-1) - k(k-1) - (n-k) (n-k-1)}{2n*(n-1) - 2k(k-1) - 2(n-k) (n-k-1)}
\]

For \( \text{Doc}(S) \), its \( P_k \) value is expected to decrease by \( \text{swap\_dis}(\text{Doc}(O), \text{Doc}(S)) \) ratio if the probability is the same for each swap’s happening.

Based on the discussions above, we can establish the relationship between Euclidean distance and precision at a certain cut-off document level: a negative correlation exists if a large number of cases are
observed. Some other measures such as average precision and recall-level precision can be addressed in a similar way. Therefore, we have demonstrated that Euclidean distance is a good effectiveness measure and conclusions drawn from Euclidean distance are also useful for other measures such as precision at a certain cut-off document level and others.

5 Previous work on data fusion

There has been quite a large body of research on data fusion in the field of information retrieval. Some early related work on data fusion is from Saracevic and Kantor [21], Turtle and Croft [23], Foltz and Dumais [8], and Belkin and his colleagues [5, 6]. Saracevic and Kantor [21] used independently-generated query representations to create a number of results, and found that a document was more likely to be relevant if it appeared in multiple results. Turtle and Croft [23] used independently-generated query representations to create a number of results within an inference network, and found that combining different query representations led to increased retrieval effectiveness over any single representation. Foltz and Dumais [8] found similar improvements by combining results from multiple retrieval strategies. Belkin and his colleagues [5, 6] conducted experiments with a 2GB TREC collection from TREC 1, and observed effectiveness improvement over a large number of combinations of different Boolean query representations. In the following we review some more work addressing certain special issues.

5.1 Data fusion methods based on score information

Fox and his colleagues [9, 10] 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.

Bartell et al. [3] 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 the linear combination method, some training queries and evaluations are required to determine the performance of those systems involved. This work demands a lot of human effort. For solving this problem, a few methods have been proposed to estimate the performance of a group of systems without any human judgment [1, 19, 22, 26].

Wu and McClean [28] proposed a group of data fusion algorithms to eliminate the effect of uneven correlation among component results by assigning different weights to all component results or their combinations. Then the linear combination method or a variation is used for fusion.

5.2 Score normalization

Usually, scores obtained from different information retrieval systems may be diverse and it is impossible to compare them directly. Therefore, some kind of score normalization is required. Linear score normalization were investigated in [14, 16].

Manmatha and colleagues [15] investigated the score distribution for a given query. They found that it might be modelled using an exponential distribution for the set of non-relevant documents and a normal distribution for the set of relevant documents. They also investigated how to use this to improve the performance of data fusion.

5.3 Data fusion methods based on ranking information

Scores are not always available. For example, very few web search engines provide scores for the retrieved web documents. Supposing that only ranking information was available, Montague and Aslam suggested a few methods including Borda fusion [2] and Condorcet fusion [17].

Using a Markov chain was investigated by some researchers [7, 20]. Markov chains can be used to realize different data fusion algorithms, which depend on the specific Markov chain involved.

5.4 Performance prediction

Vogt and Cottrell [24, 25] analyzed the performance of the linear combination algorithm by linear regression. In their experiments, they used all possible pairs of 61 systems submitted to the TREC 5 ad-hoc track. The similarity of two results' rankings and 13 other variables were used in the analysis. The performance analysis and prediction for the fused result was very accurate.

Ng and Kantor [18] used several different statistical techniques to predict if the performance of CombSum is better than both component systems involved or not. Two variables were used: performance ratio of two systems, and a measure of the dissimilarity between two systems. They found that the two variables were informative to predict if the fused result was better than both component results.

Wu and McClean [27] analyzed the performances of CombSum and CombMNZ with three groups of results submitted to TREC 6, TREC 2001 and TREC 2004. Multiple regression was used and four variables, including the number of results, the overlap rate among the results, the mean average precision of the results, and the standard deviation of the mean average precision of the results, were identified as highly significant metrics which affect the performance of data fusion.

5.5 Formal models and analysis
Lee [14] conducted an experiment to support the hypothesis: different retrieval processes might retrieve similar sets of relevant documents but retrieve different sets of non-relevant documents. Furthermore, Lee stated that as long as the component results being used for fusion had greater relevant overlap than non-relevant overlap, improvement would be observed. That can explain why the multiple evidence fusion methods such as CombMNZ are very effective data fusion methods.

Beitzel et al. [4] conducted some experiments to compare the performances of CombMNZ using several different groups of systems. They observed no improvement when fusing results from three different retrieval strategies in the same information retrieval system, while the merged result was better than the best system when choosing the top three systems submitted to TREC 6, 7, 8, 9 and 2001.

Hsu and Taksa [11] compared rank and score data fusion methods via analysis and simulation. They found that the fusion using rank information performs better than fusion using score under certain conditions.

Lalmas [13] proposed a formal model for data fusion that is based on knowledge that can be derived from the retrieved documents. Her model was based on evidential reasoning, a theory that formally expresses knowledge and the combination of knowledge.

In all these work, only Lee [14] tried to provide a condition for retrieval improvement, but which was disproved by Beitzel et. al. in [4]. To my knowledge, using the geometric probabilistic framework to describe data fusion in information retrieval is a novel approach. Under this framework, it becomes very clear why data fusion can bring retrieval improvement and what the most favourable conditions are.

6 Conclusions

In this paper we have presented a geometric probabilistic framework for data fusion in information retrieval. In this framework, we assume that each retrieved document is given an estimated relevance probability score and binary relevance judgment is applied. The key idea is to regard every component result, a list of documents, as a point in a multi-dimensional space, and then the fusion problem can be formally addressed by using geometric principles. One good characteristic is: the effectiveness of a result and the dissimilarity between two results can be measured by the same metric - the Euclidean distance. Using this framework, it is clear why data fusion can very often bring effectiveness improvement in information retrieval and what the favourable conditions are. Some useful conclusions for the fused result by centroid-based methods are summarised as follows:

(1) The fused result is at least as effective as the average of all component results in any situation;

(2) In the worst case, the effectiveness of the fused result is equal to the average effectiveness of all component results;

(3) The optimum result is possible through data fusion.

(4) The average effectiveness of all component results has a positive impact on the effectiveness of the fused result.

(5) The variance of the effectiveness of all component results has a negative impact on the effectiveness of the fused result.

(6) The average dissimilarity of all component results has a positive impact on the effectiveness of the fused result.

(7) The variance of the dissimilarity of all component results has no impact on the effectiveness of the fused result.

(8) The number of component results has a positive impact on the effectiveness of the fused result.

Since all these effects have been quantified in this paper, it thus becomes very clear how to improve the effectiveness by choosing appropriate component retrieval systems (results).

As our future work in this respect, we aim to develop an effective information retrieval system by way of fusing a group of appropriately chosen component retrieval systems. The conclusions drawn in this paper can be used as a guideline for the evaluation of the appropriateness of a group of component systems for data fusion accurately and efficiently.

References


[18] Ng, K. B., & Kantor, P. B. Predicting the effectiveness of naive data fusion on the basis of system characteristics. Journal of the American Society for Information Science, 2000, 13(50): 1177-1189.