Information Sciences 177 (2007) 3718–3730

A novel document similarity measure based on earth mover's distance

Xiaojun Wan *

Institute of Computer Science and Technology, Peking University, Beijing 100871, China

Received 14 March 2006; received in revised form 28 February 2007; accepted 28 February 2007

Abstract

In this paper we propose a novel measure based on the earth mover's distance (EMD) to evaluate document similarity by allowing many-to-many matching between subtopics. First, each document is decomposed into a set of subtopics, and then the EMD is employed to evaluate the similarity between two sets of subtopics for two documents by solving the transportation problem. The proposed measure is an improvement of the previous OM-based measure, which allows only one-to-one matching between subtopics. Experiments have been performed on the TDT3 dataset to evaluate existing similarity measures and the results show that the EMD-based measure outperforms the optimal matching (OM) based measure and all other measures. In addition to the TextTiling algorithm, the sentence clustering algorithm is adopted for document decomposition, and the experimental results show that the proposed EMD-based measure does not rely on the document decomposition algorithm and thus it is more robust than the OM-based measure.

Keywords: Document similarity measure; Document similarity search; Earth mover's distance; TextTiling; Subtopic structure

1. Introduction

Evaluating the similarity between two things is one of the most popular ways for people to compare the two things and acquire knowledge. A variety of similarity measures have been proposed for evaluating the similarity between RNA structures [6], fuzzy rulebases [14], domain concepts [8], documents [3,4,15,24], etc., respectively. Measuring pairwise document similarity is crucial for various text applications, including document clustering, document filtering, and nearest neighbor search. Most text applications aim to measure document similarity by how much information (content) two documents share. Lin [15] clarifies the intuitions about similarity as follows: The similarity between documents *a* and *b* is positively related to their commonality and negatively related to their differences. The commonality and difference between documents are usually measured based on the co-occurrences of words or phrases in the documents. If two documents share

* Tel.: +86 10 82529240; fax: +86 10 82529440.
E-mail address: wanxiaojun@icst.pku.edu.cn

0020-0255/$ - see front matter
doi:10.1016/j.ins.2007.02.045

more words/phrases while keeping less different words/phrases, the documents are more similar. Most existing similarity measures, such as the Cosine measure, the Dice measure, the Jaccard measure, the Overlap measure [4,24] and the information-theoretic measure [3], are accord with the above intuitions.

However, the above similarity measures cannot take into account document structure information (e.g., the subtopic structure[1]), and thus lose the information of word distribution over the document structure. To date, many similarity measures have been proposed to evaluate the similarity between semi-structured documents (including XML documents and HTML documents) [16,28], however, these measures cannot be applied to evaluate the similarity between plain text documents because plain text documents do not contain explicit structure information, as in the semi-structured documents. Wan and Peng [26] propose the optimal matching (OM) based similarity measure to take into account the subtopic structures of documents. The OM-based measure first uses the TextTiling algorithm [9] to decompose the documents and then applies the optimal matching theory to match the subtopics. However, the OM-based measure is limited by allowing only one-to-one matching between subtopics. Actually, any two subtopics are more or less similar, and thus they can be matched more or less. In other words, one subtopic in a document should be allowed to be matched to more than one subtopic in the other document with different weights, and thus many-to-many matching is allowed between the subtopics of two documents.

In this study we propose a novel document similarity measure based on the earth mover's distance (EMD) [20] and it allows many-to-many matching between subtopics. First, documents are decomposed into sets of subtopics, with each subtopic represented by a contiguous or incontiguous block of text. Then the earth mover's distance is employed to evaluate the similarity between the two sets of subtopics by solving the transportation problem. The experimental results on the TDT3 dataset show that the EMD-based measure outperforms all other existing similarity measures, including the OM-based measure. In addition to the TextTiling algorithm, the sentence clustering algorithm is also employed for document decomposition and the experimental results show that the EMD-based measure performs very well with either the sentence clustering algorithm or the TextTiling algorithm for document decomposition, while the OM-based measure performs poorly with the sentence clustering algorithm for document decomposition. In other words, the proposed EMD-based measure does not rely on the document decomposition algorithm, while the OM-based measure relies heavily on the TextTiling algorithm for document decomposition.

The rest of this paper is organized as follows: Section 2 reviews existing similarity measures, including the measures in the vector space model, the information-theoretic measure, the measures derived from popular retrieval functions and the OM-based measure. In Section 3, we propose the novel similarity measure based on the earth mover's distance. Experiments and results are described in Section 4. Section 5 gives our conclusions and future work.

2. Existing similarity measures

2.1. Measures in the vector space model

2.1.1. The Cosine measure

The Cosine measure is the most popular measure for evaluating document similarity based on the Vector Space Model (VSM). The VSM creates a space in which documents are represented by vectors. For a fixed collection of documents, a feature vector is generated for each document from sets of terms with associated weights. Then, a vector similarity function is used to compute the similarity between the vectors.

In the VSM, the weight $w_{d,t}$ associated with term t in any document d is calculated by $tf_{d,t} \times idf_t$, where $tf_{d,t}$ and idf_t are defined as follows:

- Term frequency $tf_{d,t}$, the number of occurrences of term t in document d.
- Inverse document frequency $idf_t = \log(N/n_t)$, where N is the total number of documents in the collection and n_t is the number of documents containing term t.

[1] In this paper, a subtopic is represented by a coherent block of text, either contiguous or incontiguous.

The similarity between two documents a and b can be defined as the normalized inner product of the two corresponding vectors $\vec{a}$ and $\vec{b}$:

$$sim_{Cosine}(a,b) = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\times\|\vec{b}\|} = \frac{\sum_{t\in a\cap b}(w_{a,t}\times w_{b,t})}{\sqrt{\sum_{t\in a}w_{a,t}^2\times\sum_{t\in b}w_{b,t}^2}} \tag{1}$$

where $a\cap b$ gets the common words between a and b. $t\in a$ (or b) means t is a unique term in document a (or b). $w_{a,t}$ and $w_{b,t}$ are the weights of term t in documents a and b, respectively.

2.1.2. The Jaccard measure

The Jaccard measure and the following two measures (i.e. the Dice measure and the Overlap measure) are all based on the VSM. The document representations are the same with the Cosine measure and the term weights are all based on the $tf_{d,t}\times idf_t$ calculation. The three measures differ from the Cosine measure in that they normalize the inner product of two document vectors in different ways. The similarity function of the Jaccard measure is as follows[2]:

$$sim_{Jaccard}(a,b) = \frac{\sum_{t\in a\cap b}(w_{a,t}\times w_{b,t})}{\sum_{t\in a}w_{a,t}^2+\sum_{t\in b}w_{b,t}^2-\sum_{t\in a\cap b}(w_{a,t}\times w_{b,t})} \tag{2}$$

2.1.3. The Dice measure

The Dice measure is defined as follows:

$$sim_{Dice}(a,b) = \frac{2\times\sum_{t\in a\cap b}(w_{a,t}\times w_{b,t})}{\sum_{t\in a}w_{a,t}^2+\sum_{t\in b}w_{b,t}^2} \tag{3}$$

2.1.4. The Overlap measure

The Overlap measure is defined as follows:

$$sim_{Overlap}(a,b) = \frac{\sum_{t\in a\cap b}(w_{a,t}\times w_{b,t})}{\min\left\{\sum_{t\in a}w_{a,t}^2,\sum_{t\in b}w_{b,t}^2\right\}} \tag{4}$$

where $\min\{x,y\}$ returns the minimal value of x and y.

2.2. The information-theoretic measure

Aslam and Frost [3] extend the concept that the assessment of pairwise object similarity can be approached in an axiomatic manner using information theory. They develop an information-theoretic measure for pairwise document similarity as follows:

$$sim_{IT\text{-}sim}(a,b) = \frac{2\times\sum_t\min\{p_{a,t},p_{b,t}\}\log\pi(t)}{\sum_t p_{a,t}\log\pi(t)+\sum_t p_{b,t}\log\pi(t)} \tag{5}$$

In the above equation, the probability $\pi(t)$ is simply the fraction of documents containing term t in the corpus. For documents a and b and term t, let $p_{a,t}$ and $p_{b,t}$ be the fractional occurrences of term t in documents a and b, respectively; thus we have $\sum_t p_{a,t}=1$ and $\sum_t p_{b,t}=1$. Two documents a and b share $\min\{p_{a,t},p_{b,t}\}$ amount of term t in common, while they contain $p_{a,t}$ and $p_{b,t}$ amount of term t individually.

2.3. Measures derived from retrieval models

2.3.1. The BM25 measure

The BM25 measure [18,19] is one of the most popular retrieval models in the probabilistic framework and it is widely used in the Okapi system. In this study, we use the BM25 model to compute the similarity value

[2] There exist variant forms of Eqs. (2)–(4) to measure the corresponding similarities and we will not list all the forms in this paper.

between documents by using one document as query. Given the query document q, the similarity score for document d is defined as follows:

$$sim_{BM25}(q,d) = \sum_{t \in q} tf_{q,t} \times \log\left(\frac{N - n_t + 0.5}{n_t + 0.5}\right) \times \frac{(K+1) \times tf_{d,t}}{K \times \left\{(1-b) + b\frac{dlf_d}{avedlf}\right\} + tf_{d,t}} \tag{6}$$

where N is the number of documents in the collection; n_t is the number of documents containing term t; $tf_{q,t}$ and $tf_{d,t}$ are the frequencies of term t in q and d, respectively; dlf_d is the sum of term frequencies in d; $avedlf$ is the average of dlf_d in the collection; K, b are parameters tuned on the training set and they are set as follows: $K = 2.0$, $b = 0.8$.

2.3.2. The vector space model with document length normalization

The vector space model with document length normalization [21,23] is a popular retrieval model used in the Smart system. Given the query document q, the similarity score for document d is defined as follows:

$$sim_{NVSM}(q,d) = \sum_{t \in q}(1 + \log(tf_{q,t})) \times idf_t \times \frac{1 + \log(tf_{d,t})}{1 + \log(avef_d)} \times \frac{1}{avedlb + S \times (dlb_d - avedlb)} \tag{7}$$

where dlb_d is the number of unique terms in d; $avef_d$ is the average of term frequencies in d (i.e. dlf_d/dlb_d); $avedlb$ is the average of dlb_d in the collection; S is a parameter tuned on the training set and it is set to 0.2.

2.3.3. The language model measure

The language model measure [7,30] adopts the probabilistic framework and it interprets the relevance between a document and a query as the probability of generating the query from the document. We use the widely used Dirichlet prior smoothing method for the unigram document model θ_d. Given the query document q, the similarity score for document d is defined as follows:

$$sim_{LM}(q,d) = \prod_{t \in q} p(t|\theta_d) = \sum_{t \in q} \log\left(\lambda \times \frac{tf_{d,t}}{dlf_d} + (1-\lambda) \times P_{MLE}(t|C)\right) \tag{8}$$

where $\lambda = dlf_d/(dlf_d + \mu)$, and $P_{MLE}(t|C)$ is the maximum likelihood estimate of the probability of term t in collection C. μ is a parameter usually set to be multiples of the average document length.

Note that the above three measures are asymmetric, i.e. given two documents a and b, the similarity value when a is taken as the query would be different from the similarity value when b is taken as the query. The three measures are widely used for keyword-based document retrieval and we can directly use them to measure document-to-document similarity in this study.

2.4. The OM-based measure

The OM-based measure is proposed by Wan and Peng [26] to take into account the subtopic structure. Given two documents a and b, the TextTiling algorithm [9] is adopted to derive their subtopic structures. Their subtopic structures are represented by the sequences of TextTiles $A = \{a_1, a_2, \ldots, a_m\}$ and $B = \{b_1, b_2, \ldots, b_n\}$ respectively, where a_i represents a subtopic (TextTile) in document a and b_j represents a subtopic (TextTile) in document b. Then a bipartite graph $G = \{A, B, E\}$ is built for documents a and b. A weight w_{ij} is assigned to every edge e_{ij}, measuring the similarity between a_i and b_j with the Cosine measure. Lastly the Kuhn–Munkres algorithm [27] is applied to acquire the total value of the optimal matching in the graph. In order to balance the effect of the lengths of different documents, the total value is normalized as follows and the normalized value is taken as the final similarity between the two documents.

$$sim_{OM}(a,b) = \frac{\omega(G)}{\min\{|A|, |B|\}} \tag{9}$$

where $\omega(G)$ represents the total value of the optimal matching in G. $|A|$ and $|B|$ represent the numbers of TextTiles in documents a and b respectively.

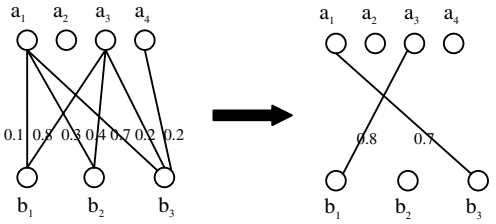

Fig. 1. An example of optimal matching.

The TextTiling algorithm will be described in detail later. The optimal matching (OM) is a classical problem in graph theory. A matching M of the bipartite graph G is a subset of the edges with the property that no two edges of M share the same node. The OM aims to find the matching M that has the largest total weight. Fig. 1 shows an OM example.

We can see that the optimal matching allows only one-to-one matching between subtopics in the documents.

The computational complexity of the Kuhn–Munkres algorithm is $O((m+n)^4)$, m, n are the numbers of vertices in the two vertex set in the bipartite graph. A faster version of the optimal matching algorithm exists [22] and it can run with the time complexity of $O((m+n)(k+(m+n)\log(m+n)))$, where k is the number of matching edges.

3. The proposed EMD-based measure

The proposed measure improves the OM-based measure by employing the earth mover's distance to allow many-to-many matching between subtopics, thus benefiting the evaluation of document similarity based on subtopic structure.

The framework of the proposed EMD-based measure is similar to the OM-based measure, consisting of the following two steps:

(1) Decompose documents into sets of subtopics;
(2) Evaluate document similarity based on the subtopic sets.

In the first step, different algorithms can be adopted to decompose documents, such as the TextTiling algorithm and the sentence clustering algorithm. In the second step, the proposed measure formalizes the comparison of two subtopic sets as the transportation problem and adopts the earth mover's distance to solve the problem, while the previous OM-based measure formalizes it as the optimal matching problem and adopts the Kuhn–Munkres algorithm to solve the problem.

3.1. Document decomposition

3.1.1. TextTiling

A document usually has its underlying discourse structure, characterized as a sequence of subtopical discussions that occur in the context of a few main topic discussions. For example, a news article about the China–US relationship, whose main topic is the good bilateral relationship between China and the United States, can be described as consisting of the following sub-discussions (initial numbers are paragraph numbers):

1 Intro-the establishment of China–US relationships;
2–3 The visits of the officers;
4–5 The culture exchange between the two countries;

6–7 The booming trade between the two countries;
8 Outlook and summary;

TextTiling is a technique for automatically subdividing text into multi-paragraph units which represent subtopics.

The TextTiling algorithm detects subtopic boundaries by analyzing patterns of lexical connectivity and word distribution. The main idea is that terms that describe a subtopic will co-occur locally, and a switch to a new subtopic will be signaled by the ending of co-occurrence of one set of terms and the beginning of the co-occurrence of a different set of terms. The algorithm consists of the following three steps:

(1) *Tokenization*: The input text is divided into individual lexical units, i.e. pseudo-sentences of a predefined size.
(2) *Lexical score determination*: All pairs of adjacent lexical units are compared and assigned with a similarity value.
(3) *Boundary identification*: The sequence of similarity values is graphed and smoothed, and then is examined for peaks and valleys. The subtopic boundaries are assumed to occur at the largest valleys in the graph.

In the algorithm, subtopic discussions are assumed to occur within the scope of one or more overarching main topics, which span the length of the text. Since the segments are adjacent and non-overlapping, they are called TextTiles.

The computational complexity is approximately linear with the document length, and more efficient implementations are available, such as Kaufmann's work [13] and JTextTile [5].

3.1.2. Sentence clustering

The clustering algorithm is often used to automatically discover the subtopics in a set of documents and group the documents into subtopic clusters. Similarly, the clustering technique can be taken to automatically discover the underlying subtopics in a set of sentences of a document and group the sentences into subtopic clusters, such as Zha's work [29]. In this study, we employ the hierarchical agglomerative clustering algorithm to group the sentences in a document and derive the subtopic structure for the document. Note that the sentences within one of the resultant subtopics might not be consecutive, while the sentences within one of the subtopics produced by the TextTiling algorithm are consecutive.

The algorithm goes as follows: initially, each sentence is considered as an individual cluster; then we iteratively merge two clusters with the largest similarity value to form a new cluster until this similarity value is below a preset merging threshold. The merging threshold can be determined through cross-validation. We employ the widely used average-link method to compute the similarity between two clusters as follows:

$$sim(c_1, c_2) = \frac{\sum_{i=1}^{p}\sum_{j=1}^{q} sim(s_i, s_j)}{p \times q} \tag{10}$$

where s_i, s_j are sentences in clusters c_1 and c_2 respectively; p is the number of sentences in cluster c_1 and q is the number of sentences in cluster c_2.

Finally, all the sentences in a cluster represent a subtopic.

The computational complexity of the clustering algorithm is $O(k^3)$, where k is the number of sentences in the document.

3.2. The EMD-based measure

The earth mover's distance (EMD) [20] is a method to evaluate dissimilarity between two multi-dimensional distributions in a feature space where a distance measure between single features, which we call the ground distance, is given. The EMD "lifts" this distance from individual features to full distributions.

Intuitively, given two distributions, one can be seen as a mass of earth properly spread in space, the other as a collection of holes in that same space. Then, the EMD measures the least amount of work needed to fill the

holes with earth. Here, a unit of work corresponds to transporting a unit of earth by a unit of ground distance. The two distributions can have different sizes.

Computing the EMD is based on a solution to the well-known transportation problem [11]. Suppose that several suppliers, each with a given amount of goods, are required to supply several consumers, each with a given limited capacity. For each supplier–consumer pair, the cost of transporting a single unit of goods is given. The transportation problem is then to find a least-expensive flow of goods from the suppliers to the consumers that satisfies the consumers' demand.

In our context, the distributions is represented by the sets of subtopics, and a weighted graph is constructed to model the similarity between two documents, and then the EMD is employed to compute the minimum cost of the weighted graph as the similarity value between two documents. The problem is formalized as follows:

Given two documents a and b, a weighted graph G is constructed as follows:

- Let $A = \{(t_{a_1}, w_{a_1}), (t_{a_2}, w_{a_2}), \ldots, (t_{a_m}, w_{a_m})\}$ as the representation of document a, t_{a_i} represents a subtopic in document a and $w_{a_i}$is the weight for t_{a_i}, calculated as the total number of the words in the sentences within subtopic t_{a_i}.
- Let $B = \{(t_{b_1}, w_{b_1}), (t_{b_2}, w_{b_2}), \ldots, (t_{b_n}, w_{b_n})\}$ as the representation of document b, t_{b_j} represents a subtopic in document b and $w_{b_j}$is the weight for t_{b_j}, calculated as the total number of the words in the sentences within subtopic t_{b_j}.
- Let $\mathbf{D} = [d_{ij}]$ as the ground distance matrix where d_{ij} is the ground distance between subtopics t_{a_i} and t_{b_j}. In our case, d_{ij} is computed by $1 - sim_{Cosine}(t_{a_i}, t_{b_j})$, where $sim_{Cosine}(t_{a_i}, t_{b_j})$ is the Cosine similarity between the two texts of subtopics t_{a_i} and t_{b_j}.
- Let $G = \{A, B, \mathbf{D}\}$ as the weighted graph constructed by A, B and $\mathbf{D}$. $V = A \cup B$ is the vertex set.

In the weighted graph G, we want to find a flow $\mathbf{F} = [f_{ij}]$, where f_{ij} is the flow between t_{a_i} and t_{b_j}, that minimizes the overall cost

$$\mathrm{WORK}(A, B, \mathbf{F}) = \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij} \tag{11}$$

subject to the following constraints:

$$f_{ij} \geqslant 0 \quad 1 \leqslant i \leqslant m, \quad 1 \leqslant j \leqslant n \tag{12}$$

$$\sum_{j=1}^{n} f_{ij} \leqslant w_{a_i} \quad 1 \leqslant i \leqslant m \tag{13}$$

$$\sum_{i=1}^{m} f_{ij} \leqslant w_{b_j} \quad 1 \leqslant j \leqslant n \tag{14}$$

$$\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} w_{a_i}, \sum_{j=1}^{n} w_{b_j}\right) \tag{15}$$

Constraint (12) allows moving words from A to B and not vice versa. Constraint (13) limits the amount of words that can be sent by any subtopic in A to its weight. Constraint (14) limits any subtopic in B to receive no more words than its weight. Constraint (15) forces to move the maximum amount of words possible. We call this amount the total flow. Once the transportation problem is solved, and we have found the optimal flow $\mathbf{F}$, the earth mover's distance is defined as follows:

$$\mathrm{EMD}(A, B) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}} \tag{16}$$

The normalization factor is introduced in order to avoid favoring shorter documents in the case of partial matching. Finally, the similarity between documents a and b is defined as follows:

$$sim_{EMD}(a, b) = 1 - \mathrm{EMD}(A, B) \tag{17}$$

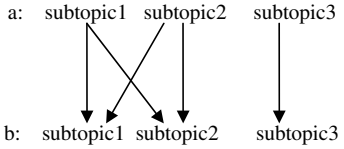

Fig. 2. An illustration example of many-to-many matching.

$sim_{EMD}(a, b)$ is normalized in the range of $[0, 1]$. The higher the value of $sim_{EMD}(a, b)$ is, the more similar documents a and b are.

The EMD-based similarity measure has the following advantages:

It can naturally extend the notion of a similarity distance between subtopics to that of a similarity distance between subtopic sets, or the whole documents. In particular, it can

- Allow for many-to-many matching between subtopics according to the ground distance matrix.
- Allow for partial matches between subtopics in a very natural way.

Fig. 2 shows an illustration example of many-to-many matching between documents a and b. The words in subtopic 1 of document a can flow to both subtopics 1 and 2 in document b. The words in subtopic 2 of document a can also flow to both subtopics 1 and 2 in document b.

Efficient algorithms for solving the transportation problem are available, and they can improve the efficiency of computing the EMD. In this study, we use the transportation simplex method [10], a streamlined simplex algorithm that exploits the special structure of the transportation problem. A theoretical analysis of the computational complexity of the transportation simplex is hard, since it is based on the simplex algorithm which can have in general an exponential worst case. However, in our context, the performance is improved by the fact that the size of the vertex set in the graph is very small. Other efficient methods to solve the transportation problem have been developed, such as the interior-point algorithms [12], which have polynomial time complexity.

4. Experiments

4.1. Experimental setup

In the experiments, a few selected similarity measures are compared with the EMD-based measure. Furthermore, the EMD-based measure and the OM-based measure are compared based on different subtopic structures derived by either the TextTiling algorithm or the sentence clustering algorithm. We evaluate the similarity measures using a document similarity search system, which ranks the documents by their similarity values with the query document.

4.1.1. Dataset

To the best of our knowledge, there has been no standard dataset for evaluating document similarity search. So we built a ground truth dataset from the TDT3 corpus, which has been used for evaluation of the task of topic detection and tracking [2] in 1999 and 2000. The TDT3 corpus was annotated by Linguistic Data Consortium (LDC) from 8 English sources and 3 Mandarin sources for the period of October through December 1998. One hundred and twenty topics were defined and about 9000 stories were annotated over these topics with an "on-topic" table presenting all stories explicitly marked as relevant to a given topic.

According to the TDT specification, the on-topic stories within the same topic were similar and relevant. After removing the stories written in Chinese, we used 40 topics as test set, and the others were used as training set for tuning the parameters in the aforementioned measures. The first document within each topic was considered as the query document and all the other documents within the same topic were the relevant (similar) documents, while all the documents within other topics were considered as irrelevant (dissimilar) documents. Each query document was compared with all documents in the whole document collection using one of the

similarity measures, and a ranked list of 500 documents was returned. The higher the document was in the ranked list, the more similar it was with the query document.

In the preprocessing step, sentence tokenization was applied to each document and the stopwords defined in the Smart system were removed from the documents. Then we used the Porter's stemmer [17] to remove common morphological and inflectional endings from English words. The TextTiling algorithm and the sentence clustering algorithm were respectively adopted to decompose documents. For the TextTiling algorithm, the JTextTile tool with the recommended parameter settings was used to segment texts into contiguous subtopic segments.

4.1.2. Evaluation metric

As in the TREC experiments [25] and previous works [1,26], we adopted the precision (P) at top N results and the non-interpolated mean average precision (MAP) to measure the performance. The metrics have been widely used to evaluate the performances of text retrieval systems.

The precision at top N results for one query was calculated as follows:

$$P@N = \frac{|C \cap R|}{|R|} \tag{18}$$

where R was the set of top N similar documents returned by the system, and C was the set of relevant documents annotated manually for the query. Then the values were averaged over all queries.

The non-interpolated average precision for one query was a number averaged over all precision values calculated after each relevant document was retrieved. If a relevant document was not retrieved, the corresponding precision value was 0.0. The values were then averaged over all queries to get the MAP value.

In the experiments, we used $P@5$, $P@10$ and MAP for evaluation, because they have been widely used for evaluation of document or Web retrieval. Note that the number of documents within each topic was different and some topics contained even less than 5 documents, so its corresponding precision values might be low. But these circumstances had no influence on the performance comparison of different measures.

4.2. Experimental results

4.2.1. Similarity measure comparison

The values of MAP, $P@5$ and $P@10$ for different similarity measures are shown and compared in Fig. 3 and Table 1. The EMD-based measure and the OM-based measure rely on the document decomposition algorithm and their performances in Fig. 3 and Table 1 are achieved based on the TextTiling algorithm. The upper bounds are the ideal values under the assumption that all the relevant (similar) documents are retrieved and ranked higher than the irrelevant (dissimilar) documents in the returned list. If the number of relevant documents for a query document is smaller than 5 or 10, the $P@5$ or $P@10$ values for this query will never reach

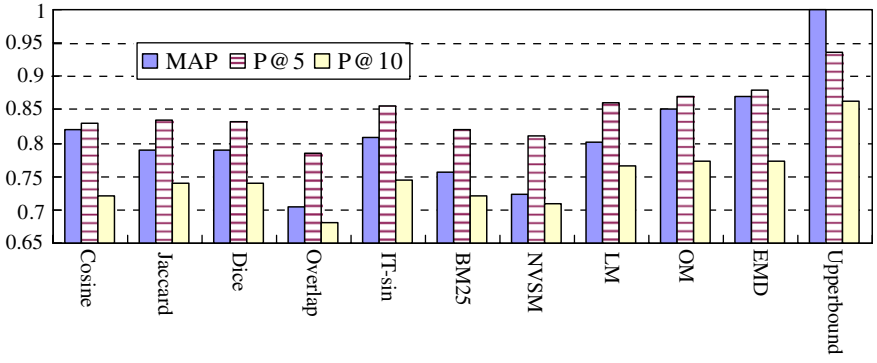

Fig. 3. Performance comparison for different similarity measures.

Table 1
Performance values for different similarity measures

	Cosine	Jaccard	Dice	Overlap	IT-sim	BM25	NVSM	LM	OM	EMD	Upperbound
MAP	0.820	0.790	0.789	0.705	0.809	0.757	0.723	0.802	0.850	**0.870**	1.000
$P@5$	0.830	0.835	0.832	0.785	0.855	0.820	0.810	0.860	0.870	**0.880**	0.935
$P@10$	0.720	0.740	0.740	0.680	0.745	0.720	0.710	0.765	**0.773**	**0.773**	0.863

100%. There are a few such queries in the TDT3 corpus, so the average $P@5$ or $P@10$ value (i.e. the upper bounds of $P@5$ or $P@10$) will not reach 100%.

Seen from Fig. 3 and Table 1, the EMD-based measure outperforms all other similarity measures over the MAP and $P@5$ metrics, including the OM-based measure (t-test: p-value < 0.02 for MAP and $P@5$).

For the measures based on the vector space model (i.e. the Cosine measure, the Jaccard measure, the Dice measure and the Overlap measure), the Cosine measure achieves the highest MAP value and comparable $P@5$ and $P@10$ values with the Jaccard measure and the Dice measure. The Overlap measure performs worst over all three metrics. The results validate the good ability of the Cosine measure for evaluating pairwise document similarity.

4.2.2. Influence of document decomposition algorithm

The proposed EMD-based measure and the OM-based measure rely on the subtopic structures, which can be derived by either the TexTiling algorithm or the sentence clustering algorithm described earlier. Figs. 4–6 show the MAP, $P@5$, $P@10$ results for the EMD-based measure and the OM-based measure with different subtopic structures, respectively. The TextTiling algorithm and the sentence clustering algorithm with different

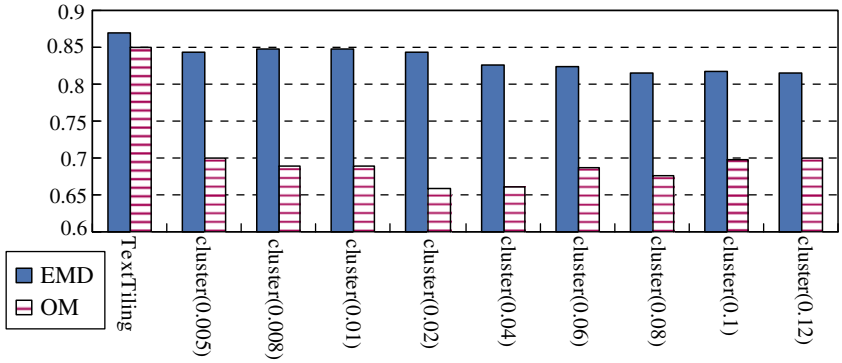

Fig. 4. MAP comparison for EMD and OM with different subtopic structures.

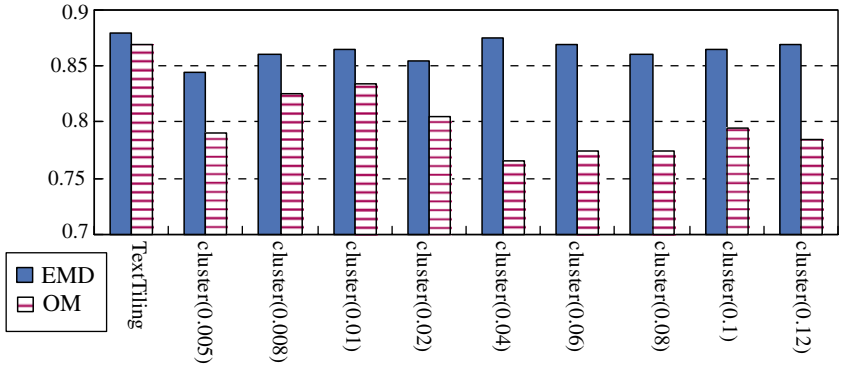

Fig. 5. $P@5$ comparison for EMD and OM with different subtopic structures.

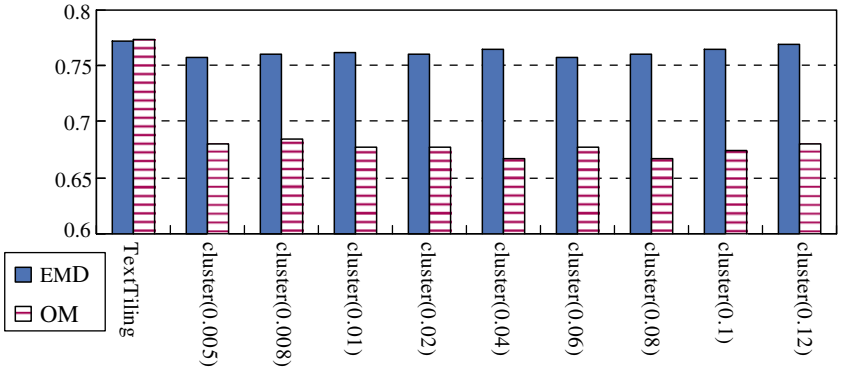

Fig. 6. $P@10$ comparison for EMD and OM with different subtopic structures.

merging similarity thresholds are explored to derive different subtopic structures. For example, "cluster(0.01)" indicates that the subtopic structure is derived by the sentence clustering algorithm with the merging similarity threshold set to 0.01.

We can see from Figs. 4–6 that the EMD-based measure performs much better than the OM-based measure when using the sentence clustering algorithm with any merging similarity threshold for document decomposition, though both measures have better performances based on the TextTiling algorithm. Moreover, the EMD-based measure can almost always perform very well based on different document decomposition algorithms, while the OM-based measure relies heavily on the TextTiling algorithm for document decomposition. In other words, the EMD-based measure is much more robust than the OM-based measure.

4.3. Discussion

First, we give some analysis of why the EMD-based measure can outperform the OM-based measure. As stated earlier, the OM allows only one-to-one matching between subtopics, and though the OM can achieve a globally optimal solution for one-to-one matching, the mappings between subtopics are not always appropriate because a subtopic with more information might correspond to another subtopic with less information. The EMD aims to address this problem by allowing many-to-many matching between subtopics under strict constraints. A subtopic is allowed to correspond to more than one subtopic and consume all the information by delivering portions of the information to different subtopics, which can help to evaluate the document similarity more appropriately.

Second, we empirically analyze the time complexity of the similarity measures. We compare the execution time of the similarity computations for pairwise documents using the Cosine measure, the OM-based measure and the EMD-based measure. The I/O time is not considered because it relies heavily on the implementation of the search engine and it is the same for all similarity measures. The execution time of each document similarity computation is recorded and then averaged. Table 2 shows the average execution time of document similarity computations using the three measures, respectively. The experiments are run on a PC with a 2.4 GHz Pentium IV processor and a maximum of 512 M Memory. The operating system is Windows 2003 Server and the programming language is Visual C#.

Seen from Table 2, the EMD-based measure has similar time complexity with the OM-based measure. Both of them are of higher time complexity than the Cosine measure, because both the OM and the EMD need to build the bipartite graph and solve a relatively more complex mathematic problem.

Table 2
Average execution time of document similarity computation

	Cosine	OM	EMD
Average execution time per document similarity computation (milliseconds)	1.48	5.16	5.44

Fortunately, we can improve the efficiency of the EMD-based measure by employing the following two approaches:

(1) We can employ advanced algorithms instead of the simplex algorithm to solve the transportation problem, such as the interior-point algorithms [12]. The algorithms have polynomial time complexity, much lower than that of the simplex algorithm, and thus they can improve the retrieval efficiency.
(2) We can employ a re-ranking process to reduce the times of the computation of the EMD. First, the Cosine measure is employed to obtain a small candidate set $C_{\text{candidate}}$ of similar documents from the whole document collection C_{all}. Then, the EMD-based measure is employed to re-rank the documents in the small set $C_{\text{candidate}}$ to get more accurate results. Seen from Table 2, the Cosine measure is much more efficient than the EMD-based measure and it can retrieve the small set of candidate documents much more quickly. Because the size of the candidate set is much smaller than the original size of the whole document collection, i.e. $|C_{\text{candidate}}| \ll |C_{\text{all}}|$, the count of the EMD computations in the re-ranking process is largely reduced from $|C_{\text{all}}|$ to $|C_{\text{candidate}}|$, which will much improve the efficiency of the whole retrieval process. In more detail, given a query, the retrieval time for the re-ranking approach is approximately $|C_{\text{all}}| \times T_{\text{Cosine}} + |C_{\text{candidate}}| \times T_{\text{EMD}}$, which is much less than that for the EMD-based measure, i.e. $|C_{\text{all}}| \times T_{\text{Cosine}} + |C_{\text{candidate}}| \times T_{\text{EMD}} \ll |C_{\text{all}}| \times T_{\text{EMD}}$, where $T_{\text{EMD}} \approx 5.44$ ms is the execution time of similarity computation for the EMD-based measure and $T_{\text{Cosine}} \approx 1.48$ ms is the execution time of similarity computation for the Cosine measure.

The first approach aims to reduce the time complexity of the EMD computation and the second approach aims to reduce the times of the EMD computation. The combination of the above two approaches will much improve the retrieval efficiency and give real-time response to user's query.

5. Conclusions

In this paper, a novel measure based on the earth mover's distance (EMD) is proposed to evaluate document similarity by allowing many-to-many matching between subtopics. The proposed measure can overcome the problem of the existing OM-based measure that only one-to-one matching is allowed between subtopics. We also explore different algorithms to decompose documents into subtopics. The experimental results show the effectiveness and robustness of the EMD-based measure.

In future work, we will combine the Cosine measure and the EMD-based measure in a re-ranking process to improve the efficiency of the EMD-based measure, as mentioned in Section 4.3. More ground truth data will be built for thorough evaluations. Other document decomposition algorithms will also be investigated to further demonstrate the robustness of the EMD-based measure.

References

[1] C.C. Aggarwal, P.S. Yu, On effective conceptual indexing and similarity search in text data, in: Proceedings of the 2001 IEEE International Conference on Data Mining, San Jose, CA, USA, 2001, pp. 3–10.
[2] J. Allan, J. Carbonell, G. Doddington, J.P. Yamron, Y. Yang, Topic detection and tracking pilot study: final report, in: Proceedings of DARPA Broadcast News Transcription and Understanding Workshop, 1998, pp. 194–218.
[3] J.A. Aslam, M. Frost, An information-theoretic measure for document similarity, in: Proceedings of the 26th International ACM/SIGIR Conference on Research and Development in Information Retrieval, 2003, pp. 449–450.
[4] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, ACM Press and Addison Wesley, 1999.
[5] F. Choi, JTextTile: a free platform independent text segmentation algorithm. <http://www.cs.man.ac.uk/~choif>.
[6] G.D. Collins, S. Le, K. Zhang, A new algorithm for computing similarity between RNA structures, Inform. Sci. 139 (1–2) (2001) 59–77.
[7] B. Croft, J. Lafferty, Language Modeling for Information Retrieval, Kluwer Academic Publishers, 2003.
[8] A. Formica, Ontology-based concept similarity in formal concept analysis, Inform. Sci. 176 (18) (2006) 2624–2641.
[9] M.A. Hearst, Multi-paragraph segmentation of expository text, in: Proceedings of the 32nd Meeting of the Association for Computational Linguistics (ACL'1994), Los Cruces, NM, 1994, pp. 9–16.
[10] F.S. Hillier, G.J. Liberman, Introduction to Mathematical Programming, McGraw-Hill, 1990.
[11] F.L. Hitchcock, The distribution of a product from several sources to numerous localities, J. Math. Phys. 20 (1941) 224–230.

[12] N. Karmarkar, A new polynomial-time algorithm for linear programming, in: Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing, 1984, pp. 302–311.
[13] S. Kaufmann, Cohesion and collocation: using context vectors in text segmentation, in: Proceedings of the 37th conference on Association for Computational Linguistics, 1999, pp. 591–595.
[14] H. Li, S. Dick, A similarity measure for fuzzy rulebases based on linguistic gradients, Inform. Sci. 176 (20) (2006) 2960–2987.
[15] D. Lin, An information-theoretic definition of similarity, in: Proceedings of the 15th International Conference on Machine Learning, 1998, pp. 296–304.
[16] A. Nierman, H.V. Jagadish, Evaluating structural similarity in XML documents, in: Proceedings of the Fifth International Workshop on the Web and Databases, 2002, pp. 61–66.
[17] M.F. Porter, An algorithm for suffix stripping, Program 14 (3) (1980) 130–137.
[18] S. Robertson, S. Walker, Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval, in: Proceedings of the 17th International ACM/SIGIR Conference on Research and Development in Information Retrieval, 1994, pp. 232–241.
[19] S. Robertson, S. Walker, M. Beaulieu, Okapi at TREC-7: automatic ad hoc, filtering, VLC and filtering tracks, in: Proceedings of TREC'99, 1999, pp. 253–264.
[20] Y. Rubner, C. Tomasi, L. Guibas, The Earth Mover's Distance as a metric for image retrieval, Int. J. Comput. Vision 40 (2) (2000) 99–121.
[21] G. Salton, The SMART Retrieval System: Experiments in Automatic Document Processing, Prentice-Hall, 1991.
[22] A. Schrijver, Combinatorial Optimization: Polyhedra and Efficiency, Volume A, Springer, Berlin, 2003.
[23] A. Singhal, C. Buckley, M. Mitra, Pivoted document length normalization, in: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996, pp. 21–29.
[24] C.J. van Rijsbergen, Information Retrieval, Butterworth, London, 1979.
[25] E. Voorhees, Overview of TREC 2001, in: Proceedings of the Tenth Text Retrieval Conference (TREC2001), 2001, pp. 1–15.
[26] X.J. Wan, Y.X. Peng, A new retrieval model based on TextTiling for document similarity search, J. Comput. Sci. Technol. 20 (4) (2005) 552–558.
[27] W.S. Xiao, Graph Theory and Its Algorithms, Aviation Industry Press, Beijing, 1993.
[28] C.C. Yang, N. Liu, Measuring similarity of semi-structured documents with context weights, in: Proceedings of SIGIR2006, pp.719–720.
[29] H. Zha, Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering, in: Proceedings of the 25th SIGIR Conference, 2002, pp. 113–120.
[30] C. Zhai, J. Lafferty, A study of smoothing methods for language models applied to ad hoc information retrieval, in: Proceedings of the 24th SIGIR Conference, 2001, pp. 334–342.