Available online at www.sciencedirect.com

ScienceDirect

Expert Systems with Applications 35 (2008) 1451–1462

Expert Systems with Applications

www.elsevier.com/locate/eswa

iSpreadRank: Ranking sentences for extraction-based summarization using feature weight propagation in the sentence similarity network

Jen-Yuan Yeh [a,*], Hao-Ren Ke [b,c], Wei-Pang Yang [d]

[a] *Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan*
[b] *Institution of Information Management, National Chiao Tung University, Hsinchu 300, Taiwan*
[c] *University Library, National Chiao Tung University, Hsinchu 300, Taiwan*
[d] *Department of Information Management, National Dong Hwa University, Hualien 974, Taiwan*

Abstract

Sentence extraction is a widely adopted text summarization technique where the most important sentences are extracted from document(s) and presented as a summary. The first step towards sentence extraction is to rank sentences in order of importance as in the summary. This paper proposes a novel graph-based ranking method, *iSpreadRank*, to perform this task. iSpreadRank models a set of topic-related documents into a sentence similarity network. Based on such a network model, iSpreadRank exploits the spreading activation theory to formulate a general concept from social network analysis: the importance of a node in a network (i.e., a sentence in this paper) is determined not only by the number of nodes to which it connects, but also by the importance of its connected nodes. The algorithm recursively re-weights the importance of sentences by spreading their sentence-specific feature scores throughout the network to adjust the importance of other sentences. Consequently, a ranking of sentences indicating the relative importance of sentences is reasoned. This paper also develops an approach to produce a generic extractive summary according to the inferred sentence ranking. The proposed summarization method is evaluated using the DUC 2004 data set, and found to perform well. Experimental results show that the proposed method obtains a ROUGE-1 score of 0.38068, which represents a slight difference of 0.00156, when compared with the best participant in the DUC 2004 evaluation.

Keywords: Sentence extraction; Multidocument summarization; Spreading activation; Sentence similarity network; Feature weigh propagation; Social network analysis

1. Introduction

The increasing amount of information has led to information overload, implying that *finding* and *using* information efficiently and effectively has become a pressingly practical problem. Search engines (e.g., Google, MSN Search, etc.) can facilitate the discovery of information by retrieving documents which are relevant to a user query. Other useful tools, such as systems that can automatically digest information content, are also desirable in processing information and making decisions.

An acute need for text summarization has emerged because of information overload (Barzilay, McKeown, & Elhadad, 1999). Text summarization refers to the process of taking a textual document, extracting content from it, and presenting the most important content to the user in a condensed form and in a manner sensitive to the user's or application's needs (Mani, 2001). The technology potentially eases the burden of information overload, since, instead of a full textual document, only a brief summary needs to be read. For instance, by providing snippets of

* Corresponding author. Tel.: +886 3 571 2121x56647; fax: +886 3 572 1490.

E-mail addresses: jyyeh@cis.nctu.edu.tw (J.-Y. Yeh), claven@lib.nctu.edu.tw (H.-R. Ke), wpyang@mail.ndhu.edu.tw (W.-P. Yang).

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2007.08.037

text for each match returned in a query, search engines can significantly help users identify preferred documents in a short time.

Text summarization was first studied in the late 1950s. Early works were based on the use of heuristics, such as term frequency (Luhn, 1958), lexical cues (Edmundson, 1969) and sentence location (Edmundson, 1969). Research in the late 1970s and the 1980s turned to complex text processing by exploiting techniques from artificial intelligence, including logic and production rules (Fum, Guida, & Tasso, 1985), scripts (Lehnert, 1982) and semantic networks (Reimer & Hahn, 1988). Dominant approaches since the 1990s have concentrated on finding characteristic text units with information retrieval and hybrid approaches (Hovy & Lin, 1997; Salton, Singhal, Mitra, & Buckley, 1997). Numerous large-scale competitions (e.g., SUMMAC,[1] DUC,[2] and NTCIR[3]) and workshops have been run to measure the performance of summarization systems as well.

This paper discusses work on multidocument summarization to create a generic extractive summary of multiple documents on the same (or related) topic. As noted in Radev, Hovy, and McKeown (2002), multidocument summarization is *the process of producing a single summary of a set of related documents* where three major issues must be addressed: (1) identifying important similarities and differences among documents; (2) recognizing and coping with redundancy, and (3) ensuring summary coherence. Previous works have investigated various techniques in solving these issues. Section 2 presents a general overview of the current state of the art.

The proposed approach adopts a broadly used summarization model – *sentence extraction* – to extract important sentences and compose them into a summary. This approach divides the multidocument summarization task into three subtasks: (1) *ranking sentences according to their importance of being part in the summary*; (2) *eliminating redundancy while extracting the most important sentences*, and (3) *organizing extracted sentences into a summary*.

This paper presents a novel sentence ranking method to perform the first subtask. The idea of a text relationship map (Salton et al., 1997) is extended to model a set of topic-related documents as a sentence-based network, based on which a graph-based sentence ranking algorithm, *iSpreadRank*, is proposed. iSpreadRank adopts a general concept from social network analysis (Carrington, Scott, & Wasserman, 2005) that the importance of a node in a network (i.e., a sentence in this paper) is not only determined by the number of nodes to which it connects, but also by the importance of its connected nodes. Specifically, iSpreadRank supposes a sentence that connects to other important sentences is itself likely to be important.

[1] http://www-nlpir.nist.gov/related_projects/tipster_summac/.

[2] http://duc.nist.gov/.

[3] http://research.nii.ac.jp/ntcir/.

iSpreadRank practically applies the spreading activation theory (Quillian, 1968) to recursively re-weight the importance of sentences by spreading their sentence-specific feature scores[4] throughout the network to modify the importance of other sentences. Consequently, a ranking of sentences indicating the relative importance of sentences is reasoned. The inferred sentence ranking is the input to other subtasks for sentence extraction.

In the second subtask, a strategy of redundancy filtering, based on cross-sentence information subsumption (Radev, Jing, Styś, & Tam, 2004), is utilized to extract one sentence at a time to the summary, if it is not too similar to any sentences already included in the summary. Finally, in the third subtask, a simplified version of the augmented sentence ordering algorithm (Barzilay, Elhadad, & McKeown, 2002) is employed to organize extracted sentences into a coherent summary.

This paper is structured as follows. Section 2 introduces the current state of the studies on multidocument summarization. While Section 3 presents an overview of the proposed summarization system, Section 4 describes the technical details of the proposed sentence ranking algorithm. The experimental results are reported in Section 5. Section 6 provides discussions on the proposed method. Finally, Section 7 concludes this paper.

2. Previous works

2.1. Overview of methods to multidocument summarization

McKeown and Radev (1995) pioneered work on multidocument summarization. They established relationships between news stories by aggregating similar extracted templates using logical relationships, such as agreement and contradiction. The summary was constructed by a sentence generator based on the facts and their relationships in the templates. These template-based methods are still of interests recently (Harabagiu & Maiorano, 2002; White et al., 2001), but require manual efforts to define domain-specific templates, while poorly-defined templates can lead to incomplete extraction of facts.

Most recent studies have adopted clustering to identify *themes*[5] (i.e., clusters) of common information (Barzilay et al., 1999; Daniel, Radev, & Allison, 2003; Goldstein, Mittal, Carbonell, & Kantrowitz, 2000; McKeown et al., 1999). These approaches are founded on an observation that multiple documents concerning a particular topic tend to contain redundant information in addition to informa-

[4] The sentence-specific feature scores work as the local information of every sentence, and are considered together with relationships between sentences to help obtain global information of sentences (i.e., the relative importance of sentences).

[5] A theme, also called a subtopic, is defined as a group of passages (such as sentences and paragraphs) that all convey approximately the same (or similar) information (McKeown, Klavans, Hatzivassiloglou, Barzilay, & Eskin, 1999).

tion unique to each document (Daniel et al., 2003). Once themes have been recognized, a representative passage in each theme is selected and included in the summary; alternatively, repeated phrases from clusters are exploited to generate an abstract-like summary by information fusion (Radev et al., 2002).

Typical research on theme clustering is briefed as follows. Barzilay et al. (1999) and McKeown et al. (1999) discovered common themes using graph-based clustering. Similar phrases in the identified themes were synthesized into a summary by information fusion. Goldstein et al. (2000) grouped paragraphs into clusters and collected in the summary from each group a significant passage with large coverage and low redundancy measured by Maximal Marginal Relevance (Carbonell & Goldstein, 1998). Daniel et al. (2003) evaluated several policies for choosing indicative sentences from sentence clusters and concluded that the best policy is to extract sentences with the highest sum of relevance scores for each cluster.

Other studies have applied information retrieval and statistical methods to find salient concepts as well as informative words and phrases in multiple documents (Harabagiu & Lacatusu, 2005; Lin & Hovy, 2002; Radev et al., 2004). For instance, Radev et al. (2004) detected a set of statistically important words as the topic centroid of a document cluster, which was treated as a feature and considered together with other heuristics to extract sentences. Lin and Hovy (2002) recognized key concepts by calculating likelihood ratios of unigrams, bigrams and trigrams of terms. Each sentence in the document set was ranked using the key concept structures in order to produce an extractive summary.

Surface-level features extended from the well-developed single-document summarization methods have also been exploited (Maña-López, Buenaga, & Gómez-Hidalgo, 2004; McDonald & Chen, 2006; Radev et al., 2004). Heuristics-based approaches selectively combine features to yield a scoring function for the discrimination of salient text units. Commonly used heuristic features include sentence position, sum of TF-IDF in a sentence, similarity with headline, sentence cluster similarity, etc.

Techniques depending on a thorough analysis of the discourse structure of the text have been explored (Chen, Wang, & Liu, 2005; Zhang, Blair-Goldensohn, & Radev, 2002). Zhang et al. (2002) developed a Cross-document Structure Theory (CST) to define the cross-document rhetorical relationships between sentences across documents. The cohesion of extractive summaries was found to be meliorated by the CST relationships. Chen et al. (2005) built lexical chains to identify topics in the input texts. Sentences were ranked according to the number of word co-occurrences in the chains and sentences.

Researchers have also investigated graph-based approaches. Mani and Bloedorn (1999) modeled term occurrences as a graph using cohesion relationships. The similarities and differences in documents were successfully pinpointed by applying spreading activation and graph matching. Some graph-based methods employ the concept of centrality in social network analysis. Salton et al. (1997) first attempted such an approach for single-document summarization. They proposed a text relationship map to represent the structure of a document, and utilized degree centrality to measure the importance of sentences.

Later works following the idea of graph-based document models employed distinct ranking algorithms to determine the centralities of sentences. Erkan and Radev (2004) recognized the most significant sentences by a sentence ranking algorithm, LexRank, which performs PageRank (Brin & Page, 1998) on a sentence-based network according to the hypothesis that sentences similar to many other sentences are salient. Erkan (2006) examined the ability of biased PageRank to extract the topic-sensitive structure beyond the text graph for question-focused summarization. Mihalcea (2004) examined several graph ranking methods originally proposed to analyze webpage prestige, including PageRank and HITS (Kleinberg, 1999), for single-document summarization. Mihalcea and Tarau (2005) extended the algorithm of Mihalcea (2004) for multiple documents. A meta-summary of documents was produced from a set of single-document summaries in an iterative manner. Zhang, Sun, and Zhou (2005) proposed a cue-based hub-authority approach that brings surface-level features into a hub/authority framework. HITS was applied in their work to rank sentences.

2.2. Comparison between graph-based related works and this work

Most graph-based methods (e.g., Erkan & Radev, 2004; Mihalcea & Tarau, 2005; Zhang et al., 2005) assess the centralities of sentences using graph-based ranking algorithms originally developed to analyze webpage prestige, including PageRank (Brin & Page, 1998) and HITS (Kleinberg, 1999). Conversely, the proposed iSpreadRank borrows concepts from the spreading activation theory (Quillian, 1968) that originated in psychology to explain the cognitive process of human comprehension. iSpreadRank further considers sentence-specific feature scores to help estimate the importance of sentences, while related works are only based on relationships between sentences (i.e., the network structure).

The use of sentence-specific features in this work resembles that of Zhang et al. (2005). However, this work is quite distinct from theirs due to the underlying ranking algorithm and the summary generation strategy. Erkan and Radev (2004) also made use of heuristic features. Different from this work, heuristic features in their work are not integrated within the ranking algorithm; instead, the graph-based centrality is viewed as another feature, and is linearly combined with other features to yield a sentence scoring function.

3. System design

Fig. 1 illustrates an overview of the proposed multidocument summarization system. The input to the system is a

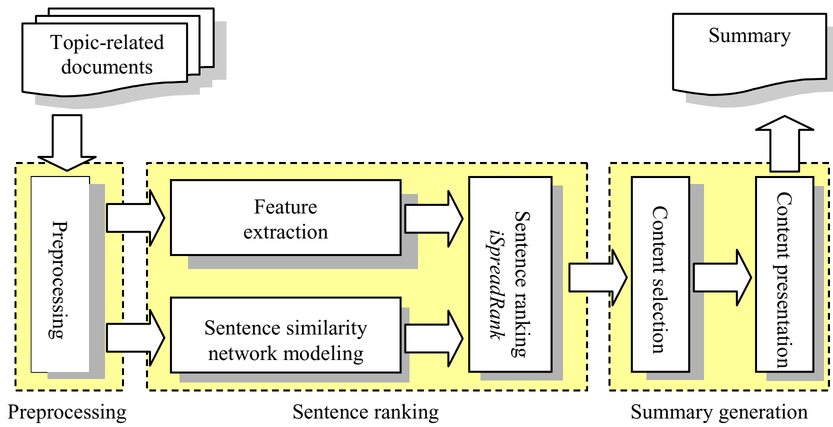

Fig. 1. System overview.

group of topic-related documents. The output is a concise summary providing the condensed essentials of the input documents. The summarizer produces an extractive summary by selecting characteristic sentences from the document group. All sentences in the document group are first ranked according to their weights of importance. Based on the ranking of sentences, the system then iteratively extracts one sentence at a time, which not only is important but also has less redundancy than other sentences extracted prior to it. The extraction finishes once the required summary length is met. The selected sentences are finally composed into the output summary.

The summarization process can be decomposed into three phases: (1) *preprocessing* preprocesses the input documents; (2) *sentence ranking* ranks the sentences according to their importance, and (3) *summary generation* creates the output summary. The entire process, as shown in Fig. 1, can be further divided into several stages, namely preprocessing, feature extraction, sentence similarity network modeling, sentence ranking, content selection and content presentation. They are outlined as follows, in order of execution:

(1) *Preprocessing:* Several linguistic analysis steps are carried out in this stage. A tokenizer segments text into words, numbers, symbols and punctuations. A sentence splitter identifies the boundaries of sentences. A passage indexer constructs a vector representation for every sentence using the well-known TF-IDF term weighting scheme (Salton & McGill, 1983).
(2) *Sentence similarity network modeling* (see Section 4.1): The input documents are transformed into a sentence-based network, with a node referring to a sentence, and an edge indicating that the corresponding sentences are related to each other. The relationship between a pair of sentences is measured by their lexical overlap.
(3) *Feature extraction* (see Section 4.2): A feature profile is created to capture the values of sentence-specific features of all sentences. Three surface-level features are employed, namely centroid, position and first-sentence overlap. The feature scores, acting as the local information of every sentence, are integrated into the proposed sentence ranking algorithm to help infer global information of sentences (i.e., the relative importance of sentences).
(4) *Sentence ranking* (see Section 4.3): A graph-based sentence ranking algorithm, *iSpreadRank*, takes a sentence similarity network and a feature profile as inputs, and applies the spreading activation theory (Quillian, 1968) to recursively re-weight the importance of sentences by spreading their sentence-specific feature scores, computed in the feature extraction stage, throughout the network. A ranking of sentences is finally inferred in order of importance.
(5) *Content selection:* A content selection module sequentially examines sentences in the rank order, and adds one sentence at a time into the summary if it is not too similar to any sentences already in the summary, as determined by a similarity threshold. This strategy only extracts high-scoring sentences with less redundant information than others based on cross-sentence information subsumption[6] (Radev et al., 2004).
(6) *Content presentation:* The final summary is structured in the following steps. Semi-similar sentences in the extracted sentence set are first grouped together, based on another similarity threshold smaller than that used in content selection. Each group is then ordered chronologically into a macro-ordering according to the earliest timestamp of the sentences within it. Finally, micro-ordering is applied to sort all sentences in each group in chronological order. This policy, considering together topical relatedness and chronological order, is a simplified form of the augmented sentence ordering algorithm (Barzilay et al., 2002).

[6] Cross-sentence information subsumption in Radev et al. (2004) was approximated using a redundancy penalty to rerank sentences; in this work, an iterative extraction process is performed instead.

4. Ranking the importance of sentences

Section 4.1 describes the modeling of a group of documents into a sentence-based network. Section 4.2 presents the extraction of sentence-specific features. Section 4.3 introduces the proposed graph-based sentence ranking algorithm, iSpreadRank.

4.1. Text as a graph: sentence similarity network

Salton et al. (1997) employed techniques for inter-document link generation to produce intra-document links between passages of a document, and obtained a text relationship map (or content similarity network). They successfully characterized the structure of a text from its linkage pattern. This work adopts the same idea to model a group of documents as a network of sentences that are related to each other, resulting in a *sentence similarity network*. A sentence similarity network is defined as a graph with nodes and edges linking nodes. Each node in the network stands for a sentence. Two sentences are connected if and only if they are similar with respect to a similarity threshold, α. In other words, an edge between two nodes indicates that the corresponding two sentences are considered to be "semantically related" (Salton et al., 1997).

This work represents each sentence as a vector of weighted terms. Let W ($|W| = n$) denote the set of terms in the document group. The vector of a sentence s_j is specified by Eq. (1), where $w_{i,j}$ is the TF-IDF weight of term t_i in s_j.

$$s_j = \langle w_{1,j}, w_{2,j}, \ldots, w_{n,j} \rangle \tag{1}$$

The degree of similarity between two sentences s_i and s_j is measured by Eq. (2) as the cosine of the angle between the vectors $\vec{s}_i$ and $\vec{s}_j$.

$$\mathrm{sim}(s_i, s_j) = \frac{\vec{s}_i \cdot \vec{s}_j}{|\vec{s}_i| \times |\vec{s}_j|} \tag{2}$$

The similarity threshold, α, is set empirically to 0.1 in the implementation.

4.2. Feature extraction

In the literature, various surface-level features have been profitably employed to determine the likelihood of sentences of being part of the summary (Kupiec, Pedersen, & Chen, 1995; Paice, 1990; Yeh, Ke, Yang, & Meng, 2005). Inspired by the success of these methods, this work attempts to integrate feature scores of sentences into the proposed graph-based sentence ranking algorithm.

This work considers three features, *centroid*, *position*, and *first-sentence overlap*, which are briefly summarized below. All of these features have been evaluated as effective predictors of the salience of sentences in Radev et al. (2004).

(1) *Centroid*: This feature measures the relatedness of a sentence and the centroid of the input document group. A sentence with more centroid words is more central to the topic.
(2) *Position*: The most important sentences tend to appear closest to the beginning of a document. This feature is computed as inversely proportional to the position of a sentence from the beginning.
(3) *First-sentence overlap*: The first sentence often introduces an overview of a document. This feature is determined as the inner-product similarity of a sentence and the first sentence in the same document.

A feature profile is generated to capture the scores of features of all sentences, and is input to the proposed sentence ranking algorithm. Each feature score in the feature profile is normalized between 0 and 1.

4.3. The proposed sentence ranking algorithm: iSpreadRank

The proposed sentence ranking algorithm, *iSpreadRank*, which is the major contribution of this work, borrows many concepts from the spreading activation theory, and is designed to rank the importance of sentences for extraction-based summarization.

S*preading activation* was originally developed in psychology to explain the cognitive process of human comprehension through semantic memory (see Quillian, 1968; Collins & Loftus, 1975; Anderson, 1983). The theory states that human long-term memory is structured as an associative network in which similar memory units have strong connections and dissimilar units have none or weak connections. Accordingly, a memory retrieval is viewed as searching across the network by activating a set of source nodes with stimuli (or energy), then iteratively propagating the energy in parallel along links through the network to other connected nodes to discover more related nodes with hidden information.

The spreading activation theory has recently been applied in many other research fields, such as information retrieval (Bollen, Vandesompel, & Rocha, 1999), hypertext structure analysis (Pirolli, Pitkow, & Rao, 1996), Web trust management (Ziegler & Lausen, 2004) and collaborative recommendation (Huang, Chen, & Zeng, 2004). This section takes the spreading activation theory one step further, and discusses combining sentence-specific feature scores and the sentence similarity network model together, under the framework of spreading activation, to reason the relative importance of sentences.

4.3.1. The algorithm

Recall that iSpreadRank supposes that the importance of a sentence is determined not only by the number of sentences to which it connects, but also by the importance of its connected sentences. In practice, iSpreadRank utilizes a particular model of spreading activation – the *Leaky Capacitor Model* (Anderson, 1983) – to realize this concept.

Adaptations are made to the model to address some practical issues.

The inputs to iSpreadRank comprise a sentence similarity network (see Section 4.1) and a feature profile (see Section 4.2). The output is a ranking of sentences indicating the importance of all sentences in order from the highest to the lowest. iSpreadRank operates in three steps: (1) *initialization*, (2) *inference*, and (3) *prediction*. The initialization step transforms the input sentence similarity network into a matrix representation for later computation. The inference step applies spreading activation to reason the relative importance of sentences, where sentence-specific local importance, initialized by the input feature profile, recursively spreads throughout the whole network. In this step, the algorithm iterates until an equilibrium state of the network is achieved. Finally, the prediction step outputs a ranking of sentences according to the inference results in the inference step.

In summary, the goal of iSpreadRank is to re-weight similar sentences with similar degree of importance, and hence rank them in close positions in the reasoned ranking.

(1) *Initialization.* Let $G = (V, E)$ represent the sentence similarity network with the set of nodes $V = \{s_1, \ldots, s_m\}$ and the set of edges E, where s_i denotes a sentence, and E is a subset of $V \times V$. For simplicity, every node with no edges connecting it to other nodes is further eliminated from G. Such a weighted graph representation of the input document group can be transformed into an adjacency matrix, A, with rows and columns labeled by sentence nodes, and each entry a_{ij} initialized by Eq. (3). Notably, A is a symmetric matrix since G is an undirected graph.

$$a_{ij} = a_{ji} = \begin{cases} 0 & \text{if } i = j \\ \mathrm{sim}(s_i, s_j) & \text{if } i \neq j \end{cases} \tag{3}$$

Here, $\mathrm{sim}(s_i, s_j)$ indicates the similarity between a pair of sentences s_i and s_j (see Eq. (2)) and $\mathrm{sim}(s_i, s_j) \geqslant \alpha$ (α is the similarity threshold mentioned in Section 4.1).

(2) *Inference.* Each node in the network has an activation level.[7] The algorithm iteratively updates the activations of all nodes over discrete time until it is stopped by the user, or a termination condition is triggered. In one iteration, each node obtains a new activation level by collecting the activations from its connected nodes, and then propagates the new activation along links to its neighbors as a function of its current activation and the relative weights between nodes.

The iteration itself can be mathematically defined in simple linear algebra. Let X represent an m-dimensional vector to capture the activations of nodes in the network. A particular vector, $X(0)$, is the activation vector at the initial step where the activation of each sentence node is initialized as its sentence-specific feature score computed by feature extraction (see Section 4.2). In iteration t, the algorithm maintains the activation vector $X(t)$ using Eq. (4)[8].

$$X(t) = X(0) + MX(t-1), \quad M = \sigma R^{\mathrm{T}} \tag{4}$$

In the equation, σ ($0 \leqslant \sigma < 1$) is a spreading factor determining the propagation efficiency to which a node converts the activations from its neighbors to its own activation (i.e., the level of activation propagated from a node's neighbors to the node). It is assigned heuristically to 0.7 in the implementation. The matrix R is obtained from A by Eq. (5). Since the Initialization step removes nodes with no edges, R is a stochastic matrix, i.e., for each row i in R, $\sum_j r_{ij} = 1$.

$$r_{ij} = \frac{a_{ij}}{\sum_k a_{ik}} \tag{5}$$

The algorithm iterates until a stable equilibrium of the network (i.e., the converged state) is obtained. Practically, a stopping condition judges the convergence of the algorithm and terminates the iterations. In this work, each iteration is followed by a checkpoint to determine whether the criterion in Eq. (6) is satisfied. In the equation, $X_i(t)$ refers to the activation of node i at step t, and ε is a negligible number, set to 0.0001 in this work. Specifically, Eq. (6) measures the L_1 norm of the residual vector: $X(t) - X(t-1)$.

$$\sum_i |X_i(t) - X_i(t-1)| \leqslant \varepsilon \tag{6}$$

The algorithm terminates at iteration t when the sum of changes of the activations for all nodes with respect to prior iteration $t-1$ is not greater than a predefined threshold ε.

(3) *Prediction.* When iSpreadRank ends, the network is in a stable state with each node labeled with a numeric weight as its final degree of importance. iSpreadRank outputs a ranking of sentences according to the importance of all sentences inferred in the inference step. (N.B. for those sentences without connections to other sentences, their initial feature scores are used for ranking.)

4.3.2. *The convergence of iSpreadRank*

The convergence of iSpreadRank is proven via Proposition 1.

[7] The term "activation" is interchangeable with the term "importance" in this context. It is used here in order to follow the terminology of spreading activation.

[8] The equation used in this work is a simplified leaky capacitor model. For an introduction of the original model and a comparison with iSpreadRank, please refer to Section 6.3.

Proposition 1. For some t, $t > 0$,

(a) $\sum_i |X_i(t) - X_i(t-1)| \leqslant \varepsilon$. $\iff$ (b) iSpreadRank converges at t-th iteration.
(b) iSpreadRank converges at t-th iteration. $\iff$ (c) $X(t) \approx (I - \sigma R^{\mathrm{T}})^{-1} X(0)$.
(a) $\sum_i |X_i(t) - X_i(t-1)| \leqslant \varepsilon$. $\iff$ (c) $X(t) \approx (I - \sigma R^{\mathrm{T}})^{-1} X(0)$.

I: (a) $\Rightarrow$ (b).

Proof. Consider $X(t+1)$ and $X(t)$. According to Eq. (4), the following equations hold:

$$X(t+1) = X(0) + \sigma R^{\mathrm{T}} X(t) \tag{I.1}$$

$$X(t) = X(0) + \sigma R^{\mathrm{T}} X(t-1) \tag{I.2}$$

Since $\sum_i |X_i(t) - X_i(t-1)| \leqslant \varepsilon$ and ε is negligible, assume $X(t) = X(t-1)$. By replacing $X(t)$ in Eq. (I.1) with $X(t-1)$, Eq. (I.3) is obtained.

$$X(t+1) = X(0) + \sigma R^{\mathrm{T}} X(t-1) \tag{I.3}$$

From Eqs. (I.2) and (I.3), $X(t+1) = X(t)$.

By induction, it is easily verified that $\forall t'$, $t' = t + c$ and $c \geqslant 0$, $X(t') = X(t'-1)$ holds. Hence, iSpreadRank converges at t-th iteration. □

II: (b) $\Rightarrow$ (a).

Proof. Since iSpreadRank converges at t-th iteration, $\forall t', t' = t + c$ and $c \geqslant 0$, $X(t') \approx X(t'-1)$ holds. Then, $\sum_i |X_i(t') - X_i(t'-1)| \leqslant \varepsilon$. □

III: (a) $\iff$ (b).

Proof. From I: (a) $\Rightarrow$ (b) and II: (b) $\Rightarrow$ (a), it is proven. □

IV: (b) $\Rightarrow$ (c).

Proof. Since iSpreadRank converges at t-th iteration, assume $X(t) = X(t-1)$. By replacing $X(t-1)$ in Eq. (4) with $X(t)$, it is easily verified that

$$(I - \sigma R^{\mathrm{T}}) X(t) = X(0).$$

Let $P = I - \sigma R^{\mathrm{T}}$, $P^{\mathrm{T}} = I - \sigma R$. Since R is a stochastic matrix and its diagonals are all 0s, and $0 \leqslant \sigma < 1$, P^{T} is a strictly diagonally dominant matrix. The Gerschgorin circle theorem (Noble & Daniel, 1988) assures that the inverse of P^{T} exists. Since $P^{\mathrm{T}} = I - \sigma R$ is invertible, $P = I - \sigma R^{\mathrm{T}}$ is also invertible and hence $X(t) = (I - \sigma R^{T})^{-1} X(0)$. □

V: (c) $\Rightarrow$ (b).

Proof. Suppose iSpreadRank does not converge at t-th iteration and assume $X(t)! \approx X(t-1)$. Similarly, by Eq. (4), it is easily verified that

$$(I - \sigma R^{\mathrm{T}}) X(t)! \approx X(0).$$

As in IV: (b) $\Rightarrow$ (c), $P = I - \sigma R^{\mathrm{T}}$ is invertible and hence $X(t)! \approx (I - \sigma R^{\mathrm{T}})^{-1} X(0)$, which is contradictory to the given $X(t) \approx (I - \sigma R^{\mathrm{T}})^{-1} X(0)$. Therefore, iSpreadRank converges at t-th iteration. □

VI: (b) $\iff$ (c).

Proof. From IV: (b) $\Rightarrow$ (c) and V: (c) $\Rightarrow$ (b), it is proven. □

VII: (a) $\iff$ (c).

Proof. From III: (a) $\iff$ (b) and VI: (b) $\iff$ (c), it is proven. □

It is guaranteed that there is a t since $(I - \sigma R^{\mathrm{T}})^{-1} X(0)$ does exist. On the basis of Proposition 1, it is proven that for such a t, Eq. (6) is satisfied (and iSpreadRank terminates) and iSpreadRank converges at t-th iteration.

4.3.3. Example

Fig. 2 illustrates the working of iSpreadRank to re-weight the importance of sentences. Fig. 2(a) displays the initial state of the network before iSpreadRank is applied.

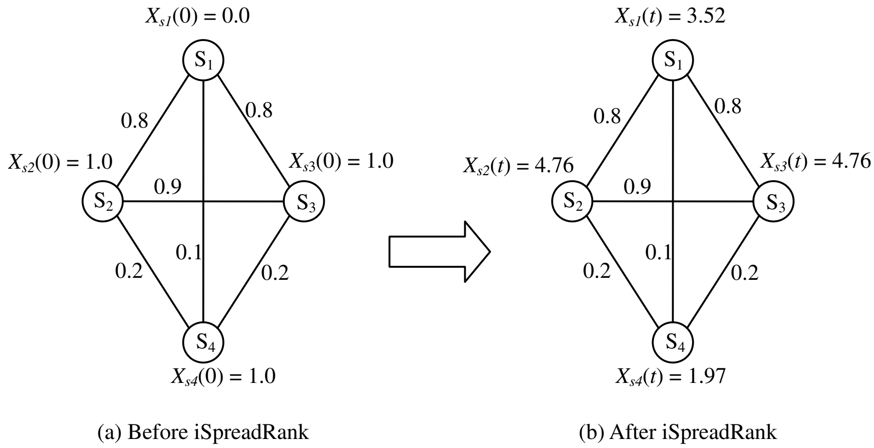

Fig. 2. An example to explain how iSpreadRank works (the spreading factor $\sigma = 0.8$). (a) The initial state of the network before iSpreadRank is applied; (b) the converged state when iSpreadRank terminates at iteration t.

Table 1
Weights of the inferred importance for S_i at different iterations (the spreading factor $\sigma = 0.8$)

Iteration	S_1	S_2	S_3	S_4
0	0.0000	1.0000	1.0000	1.0000
1	0.8337	1.6989	1.6989	1.1684
5	2.4058	3.5114	3.5114	1.6392
10	3.1543	4.3489	4.3489	1.8594
20	3.4802	4.7131	4.7131	1.9552
≈Convergence	3.5193	4.7568	4.7568	1.9667

The sentence ranking is $\text{Rank}(S_2) = \text{Rank}(S_3) = \text{Rank}(S_4) > \text{Rank}(S_1)$. Given this network, iSpreadRank runs and terminates at the converged state, as depicted in Fig. 2(b), and outputs a new sentence ranking: $\text{Rank}(S_2) = \text{Rank}(S_3) > \text{Rank}(S_1) > \text{Rank}(S_4)$. It can be seen that S_1 is promoted to the position before S_4 in the new ranking.

Table 1 presents the weights of the inferred importance of S_i at different iterations. According to this table, the weight of S_1 raises more rapidly than the weight of S_4 during the inference iterations. This is because S_1 is strongly related to S_2 and S_3, and therefore it receives more weights distributed from them. In contrast, S_2 and S_3 propagate fewer weights to S_4 since S_4 has weak connections with S_2 and S_3. Consequently, S_1 obtains a new weight, $X_{s1}(t) = 3.5193$, which is much larger than the new weight of S_4, $X_{s4}(t) = 1.9667$. Furthermore, S_1, S_2, and S_3 together form a feedback loop, giving them the highest weights in the end.

5. Evaluation

This section describes the data set, evaluation metric, and the experimental results.

5.1. Data set and experimental setup

The DUC 2004 data set from DUC (Document Understanding Conferences) was tested to examine the effectiveness of the proposed summarization method (see Fig. 1 for the system overview). The guideline of Task 2 at the DUC 2004 was followed to produce generic extractive summaries. The task is to generate a short summary of roughly 665 bytes in length to provide the condensed essentials of an input group of topic-related news articles.

The total number of document groups is 50. Each group contains 10 newswire articles on average. For each group, four NIST assessors were each asked to read all the documents and to create a brief summary. The manually-generated summaries are treated as gold-standard summaries to evaluate the qualities of machine-generated summaries.

5.2. Evaluation metric

Machine-generated summaries are evaluated using ROUGE (Recall-Oriented Understudy for Gisting Evaluation) automatic n-gram matching (Lin & Hovy, 2003). ROUGE is a recall-based scoring metric for fix-length summaries, which adopts ideas from BLEU (BiLingual Evaluation Understudy) (Papineni, Roukos, Ward, & Zhu, 2001) to determine the quality of a machine-generated summary. It generally counts as a performance indicator the number of co-occurrences between machine-generated and ideal summaries in different word units, such as n-gram, word sequences and word pairs.

The official ROUGE scores at the DUC 2004 are the *1-gram*, *2-gram*, *3-gram*, *4-gram*, and *longest common substring* scores. The 1-gram ROUGE score (a.k.a. ROUGE-1) has been found to correlate very well with human judgements at a confidence level of 95%, based on various statistical metrics (Lin & Hovy, 2003). Therefore, this paper only reports the ROUGE-1 scores.

5.3. Results

Table 2 lists the ROUGE-1 scores of different experiments and their 95% confidence intervals in brackets. *Feature* denotes which sentence-specific feature is used to

Table 2
ROUGE-1 scores of Without-iSpreadRank and With-iSpreadRank in different settings

Feature	Without-iSpreadRank	With-iSpreadRank ($\sigma = 0.7$)	Improvement
EV = 1	–	0.36218 [0.34611, 0.37825]	–
Centroid (C)	0.35033 [0.33354, 0.36712]	0.36722 [0.35308, 0.38136]	+0.0169 (4.82%)
Position (P)	0.36524 [0.35290, 0.37758]	0.37756 [0.36324, 0.39188]	+0.0123 (3.37%)
SimWithFirst (SF)	0.36524 [0.35290, 0.37758]	0.37052 [0.35903, 0.38201]	+0.0053 (1.45%)
C + P	0.36974 [0.35807, 0.38141]	0.37701 [0.36429, 0.38973]	+0.0073 (1.97%)
C + SF	0.36923 [0.35747, 0.38099]	0.37821 [0.36551, 0.39091]	+0.0090 (2.44%)
P + SF	0.36524 [0.35290, 0.37758]	0.37355 [0.36063, 0.38647]	+0.0083 (2.27%)
C + P + SF	0.37333 [0.36182, 0.38484]	0.38068 [0.36804, 0.39332]	+0.0074 (1.97%)
Random baseline:	0.31549 [0.30332, 0.32766]		
NIST baseline:	0.32419 [0.30922, 0.33916]		

estimate the importance of every sentence. *Without-iSpreadRank* scores sentences only by features, while *With-iSpreadRank* applies the proposed iSpreadRank for sentence ranking. *Improvement* refers to the difference between the ROUGE-1 scores and the relative improvement[9] in the parentheses when With-iSpreadRank is compared to Without-iSpreadRank. Table 2 also presents two baselines. *Random Baseline* randomly extracts sentences from the input document group. The reported result is averaged from 10 random runs. *NIST Baseline*, the official baseline at the DUC 2004, simply outputs the first 665 bytes of the most recent document.

Several interesting results are found. First, With-iSpreadRank performs significantly better than the two baselines. Second, With-iSpreadRank is superior to Without-iSpreadRank, which demonstrates that the use of sentence-specific features in iSpreadRank is an effective sentence ranking method. The average improvement is observed to decrease when the initial importance of sentences is determined by more features. The average improvement is 3.21% when only one feature is used, becoming 2.23% when employing two features, 1.97% when all features are examined. This phenomenon merits further investigation. Third, a particular experiment (see Feature: EV = 1) was conducted in which iSpreadRank initially assigned every sentence an equal feature score of 1.0. In this case, iSpreadRank depends much on the relationships between sentences, and ranks sentences similar to many other sentences in high positions. As expected, this model is inferior to other models where *real* sentence-specific features are considered. This result confirms that the importance of a sentence is determined not only by the number of sentences to which it connects, but also by the importance of its connected sentences.

Table 3 shows the official ROUGE-1 scores of human assessors and the top 5 systems for Task 2 at the DUC 2004. In this table, *SYSID* signifies the peer codes of participants: letters stand for human assessors, and numbers represent machine systems. The scores indicate, at the 95% confidence level, that With-iSpreadRank does not outperform the best machine (SYSID: 65) in any settings. However, four of them performed better than the second best system (SYSID: 104), namely (1) With-iSpreadRank + Feature: C + P + SF, (2) With-iSpreadRank + Feature: C + SF, (3) With-iSpreadRank + Feature: C + P and (4) With-iSpreadRank + Feature: P. Overall, the proposed summarization method is found to perform well with competitive results. The best model of With-iSpreadRank (i.e., With-iSpreadRank + Feature: C + P + SF) has a ROUGE-1 score of 0.38068, which represents a slight difference of 0.00156 in comparison with the 1st-ranked system (SYSID: 65) at the DUC 2004.

Table 3
Part of the official ROUGE-1 scores of Task 2 at the DUC 2004

SYSID	ROUGE-1	95% Confidence interval
H	0.41828	[0.40193, 0.43463]
F	0.41246	[0.39161, 0.43331]
E	0.41038	[0.38817, 0.43259]
D	0.40594	[0.38700, 0.42488]
B	0.40428	[0.37946, 0.42910]
A	0.39325	[0.37218, 0.41432]
C	0.39039	[0.37149, 0.40929]
G	0.38902	[0.36793, 0.41011]
65	0.38224	[0.36941, 0.39507]
104	0.37443	[0.36354, 0.38532]
35	0.37430	[0.36121, 0.38739]
19	0.37386	[0.36080, 0.38692]
124	0.37064	[0.35782, 0.38346]
2 (NIST Baseline) (Rank: 25/35)	0.32419	[0.30922, 0.33916]
Best machine (SYSID = 65)	0.38224	[0.36941, 0.39507]
Median machine (SYSID = 138)	0.34299	[0.32805, 0.35793]
Worst machine (SYSID = 111)	0.24190	[0.23038, 0.25342]
Avg. of human assessors	0.40300	[0.38247, 0.42353]

6. Discussions

6.1. Sentence similarity network

The major problem of a sentence similarity network constructed using the cosine similarity (as adopted in this

[9] The relative improvement is calculated as $(b - a)/a * 100$ when b is compared to a.

paper) is the lack of type or context in a link (Salton et al., 1997). Fortunately, this problem can be alleviated by considering semantic-level text analysis when defining the similarity between text units (see Hatzivassiloglou et al., 2001; Mihalcea, Corley, & Strapparava, 2006; Yeh et al., 2005). For instance, Yeh et al. (2005) found that the similarity computed from latent semantic analysis improves the performance of degree-centrality-based single-document summarization. According to their observations, we expect that the improvement of relationships between sentences will directly profit iSpreadRank. This issue is left to future work.

6.2. The use of sentence-specific features

With the use of sentence-specific features, iSpreadRank operates like a semi-supervised learning process in which the initial labeling of every sentence is determined according to its feature score, and the final labeling of sentences is achieved based on the feature scores of sentences and the relationships between sentences. This work tested three features: *centroid*, *position*, and *first-sentence overlap*, as well as various combinations of them, to understand how they affect the performance of iSpreadRank. Table 2 reveals that the performance is improved when sentence-specific features are considered.

Evaluation results in this work demonstrate that particular surface-level features that proven effective in text summarization could be profitably employed in iSpreadRank. The sentence-specific features that are advantageous to iSpreadRank are worth studying. However, this issue is left as an open question, since examining the whole feature space is not straightforward.

6.3. iSpreadRank

iSpreadRank applies a particular model of spreading activation, namely the Leaky Capacitor Model (LCM) (Anderson, 1983). LCM formulates the flow of activations of all the nodes over time by Eq. (7).[10]

$$X(t) = C + MX(t-1), \quad M = \left[(1-\gamma)I + \sigma R^{\mathrm{T}}\right] \tag{7}$$

where C indicates a vector capturing the set of energized nodes and their activations at iteration t; M represents a matrix to manage the flow and the decay of activation among nodes; $\gamma \in [0,1]$ determines the relaxation of node activation; I denotes the identity matrix, and σ and R are as in Eq. (4).

iSpreadRank is a derivative of LCM since it simply fixes $C = X(0)$ and $\gamma = 1$ in all iterations. However, iSpreadRank is very different from LCM in terms of its goal and how it is achieved. In general, LCM only activates a *subset* of nodes in each iteration; iSpreadRank, in contrast, propagates the activations of *all* nodes into the network (i.e., all nodes are activated). Additionally, while LCM is designed to identify hidden nodes related to the activated source nodes according to some criterion, the goal of iSpreadRank is to assess the relative importance of all nodes.

6.3.1. Spreading factor

The value of σ generally depends on different applications, and may be tuned after running a number of preliminary experiments. With a high value of σ, the activation of a node propagated to its neighbors is in large amount, and the activation is spread to nodes further away in iterations (Ziegler & Lausen, 2004). In this case, iSpreadRank outputs a ranking relying significantly on global information of the whole network. With a low value of σ, the propagation of activations among nodes becomes moderate, leading to an output ranking close to the initial ranking provided by the sentence scoring function based on sentence-specific features.

6.4. The proposed summarization method

The proposed summarization method has several benefits. First, it is an unsupervised approach, and therefore requires no training data. Second, the proposed method is domain-independent as well as language-independent, since it considers neither domain-specific knowledge nor deep linguistic analysis of texts. Third, the proposed method is extensible owing to its modulization design (see Fig. 1). For example, distinct surface-level features can be easily employed in iSpreadRank to help assess the importance of sentences.

The proposed method can be regarded as a theme clustering based approach. Recall that iSpreadRank re-weights similar sentences with similar degree of importance, and ranks them in close positions in the inferred ranking. Consequently, a sequence of similar sentences with close weights constitutes a partition of the ranking. Consider as well the content selection module in Fig. 1; it sequentially examines sentences in the rank order, and adds one sentence at a time into the summary if it is not too similar to any sentences already in the summary. Successive sentences after a selected sentence are thus skipped until a dissimilar sentence is found. Based on these principles, the selection of the preceding sentence (i.e., the sentence with the highest weight) in a partition is similar to the extraction of a representative sentence from a subtopic, which is a common strategy used in theme clustering based approaches.

7. Conclusion and future work

This paper proposes a novel graph-based sentence ranking method, iSpreadRank, to rank the importance of sentences for extraction-based summarization. iSpreadRank models a set of topic-related documents into a sentence

[10] This matrix calculus is excerpted from Pirolli et al. (1996) with adaptations in correspondence to the terminology used in this paper.

similarity network in which nodes denote sentences, and edges indicate the relationships between the sentences. The spreading activation theory is then applied to recursively re-weight the importance of sentences by spreading their sentence-specific feature scores throughout the network to adjust the importance of other sentences. With the use of sentence-specific features, iSpreadRank operates like a semi-supervised learning process in which the initial labeling of every sentence is determined by its feature score, and the final labeling of sentences is based on the feature scores of sentences and the relationships between them. Thus, a ranking of sentences indicating their relative importance is reasoned.

This paper also develops a method to produce an extractive generic summary of multiple documents based on the reasoned sentence ranking. To address multidocument summarization, iSpreadRank is integrated with two techniques that have been proven effective in the field of anti-redundancy and sentence ordering. The first technique is a redundancy filtering strategy based on cross-sentence information subsumption (Radev et al., 2004) to extract only high-scoring sentences with little redundant information. The second is a simplified version of the augmented sentence ordering algorithm (Barzilay et al., 2002) to organize extracted sentences into a coherent summary.

The proposed summarization method is evaluated with the DUC 2004 data set, and found to perform well. Three sentence-specific features, (1) centroid, (2) position, and (3) first-sentence overlap, were tested along with their combinations, in order to understand how they affect the performance of iSpreadRank. Experimental results demonstrate that the performance is improved when features are considered in iSpreadRank, but the average improvement decreases as more features are considered together. This issue needs to be investigated in the future. A particular experiment (see Feature: $EV = 1$ in Table 2) was also conducted in which iSpreadRank initially assigned every sentence an equal feature score of 1.0. As expected, this model is inferior to other models that consider *real* sentence-specific features. This result corresponds to the concept that the importance of a sentence is determined not only by the number of sentences to which it is connected, but also by the importance of its connected sentences. In summary, the proposed method obtains a ROUGE-1 score of 0.38068, and is ranked in the second place in the DUC 2004 evaluation.

Future work will continue to test the ability of iSpreadRank in the query-oriented summarization task where the relatedness of a sentence and the query could be regarded as another feature in iSpreadRank to discover the query-sensitive structure beyond the sentence similarity network. It should also be important to study whether the improvement of relationships between sentences in the sentence similarity network will directly profit iSpreadRank. Another interesting issue is to investigate what kinds of sentence-specific features are advantageous to iSpreadRank.

Acknowledgements

This work was supported by the National Science Council (Grant Number: NSC-92-2213-E-009-126). Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors only, and do not necessarily reflect the viewpoints of the National Science Council.

References

Anderson, J. R. (1983). A spreading activation theory of memory. *Journal of Verbal Learning and Verbal Behavior, 22*, 261–295.

Barzilay, R., Elhadad, N., & McKeown, K. R. (2002). Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research, 17*, 35–55.

Barzilay, R., McKeown, K. R., & Elhadad, M. (1999). Information fusion in the context of multi-document summarization. In *Proceedings of the 37th annual meeting of the association for computational linguistics* (pp. 550–557). College Park, MD, USA.

Bollen, J., Vandesompel, H., & Rocha, L. M. (1999). Mining associative relations from website logs and their applications to context-dependent retrieval using spreading activation. In *Proceedings of the workshop on organizing web space*. Berkeley, CA, USA.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems, 30*(1-7), 107–117.

Carbonell, J., & Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval* (pp. 335–336). Melbourne, Australia.

Carrington, P. J., Scott, J., & Wasserman, S. (Eds.). (2005). *Models and methods in social network analysis*. New York, NY: Cambridge University Press.

Chen, Y.-M., Wang, X.-L., & Liu, B.-Q. (2005). Multi-document summarization based on lexical chains. In *Proceedings of the 2005 international conference on machine learning and cybernetics* (pp. 1937–1942). Beijing, China.

Collins, A. M., & Loftus, E. F. (1975). A spreading-activation theory of semantic processing. *Psychological Review, 82*(6), 407–428.

Daniel, N., Radev, D., & Allison, T. (2003). Sub-event based multi-document summarization. In *Proceedings of the HLT-NAACL '03 workshop on text summarization* (pp. 9–16). Edmonton, Canada.

Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the ACM, 16*(2), 264–285.

Erkan, G. (2006). Using biased random walks for focused summarization. In *Proceedings of the DUC 2006 document understanding workshop*. Brooklyn, NY, USA.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research, 22*, 457–479.

Fum, D., Guida, G., & Tasso, C. (1985). Evaluating importance: A step towards text summarization. In *Proceedings of the 9th international joint conference on artificial intelligence* (pp. 840–844). Los Angeles, CA, USA.

Goldstein, J., Mittal, V., Carbonell, J., & Kantrowitz, M. (2000). Multi-document summarization by sentence extraction. In *Proceedings of the NAACL-ANLP 2000 workshop on automatic summarization* (pp. 40–48). Seattle, WA, USA.

Harabagiu, S., & Lacatusu, F. (2005). Topic themes for multi-document summarization. In *Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 202–209). Salvador, Brazil.

Harabagiu, S., & Maiorano, S. (2002). Multi-document summarization with GISTexter. In *Proceedings of the 3rd LREC conference*. Canary Islands, Spain.

Hatzivassiloglou, V., Klavans, J. L., Holcombe, M. L., Barzilay, R., Kan, M.-Y., & McKeown, K. R. (2001). SimFinder: A flexible clustering tool for summarization. In *Proceedings of NAACL workshop on automatic summarization* (pp. 41–49). Pittsburgh, PA, USA.

Hovy, E., & Lin, C.-Y. (1997). Automated text summarization in SUMMARIST. In *Proceedings of the ACL97/EACL97 workshop on intelligent scalable text summarization* (pp. 18–24). Madrid, Spain.

Huang, Z., Chen, H., & Zeng, D. (2004). Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. *ACM Transactions on Information Systems, 22*(1), 116–142.

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM, 46*(5), 604–632.

Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 68–73). Seattle, WA, USA.

Lehnert, W. G. (1982). Plot units: A narrative summarization strategy. In W. G. Lehnert & M. H. Ringle (Eds.), *Strategies for natural language processing* (pp. 375–412). Hillsdale, NJ: Lawrence Erlbaum.

Lin, C.-Y., & Hovy, E. (2002). NeATS in DUC 2002. In *Proceedings of the DUC 2002 workshop on text summarization*. Philadelphia, PA, USA.

Lin, C.-Y., & Hovy, E. (2003). Automatic evaluation of summaries using N-gram co-occurrence statistics. In *Proceedings of the 2003 conference of the North American chapter of the association for computational linguistics on human language technology* (pp. 71–78). Edmonton, Canada.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development, 2*(2), 159–165.

Maña-López, M. J., Buenaga, M. D., & Gómez-Hidalgo, J. M. (2004). Multidocument summarization: An added value to clustering in interactive retrieval. *ACM Transaction on Information Systems, 22*(2), 215–241.

Mani, I. (2001). *Automatic summarization*. Amsterdam, The Netherlands: John Benjamins Pub Co.

Mani, I., & Bloedorn, E. (1999). Summarizing similarities and differences among related documents. *Information Retrieval, 1*(1–2), 35–67.

McDonald, D. M., & Chen, H. (2006). Summary in context: Searching versus browsing. *ACM Transactions on Information Systems, 24*(1), 111–141.

McKeown, K. R., Klavans, J. L., Hatzivassiloglou, V., Barzilay, R., & Eskin, E. (1999). Towards multidocument summarization by reformulation: Progress and prospects. In *Proceedings of the 16th national conference on artificial intelligence* (pp. 453–460). Orlando, FL, USA.

McKeown, K., & Radev, D. R. (1995). Generating summaries of multiple news articles. In *Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval* (pp. 74–82). Seattle, WA, USA.

Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the 42nd annual meeting of the association for computational linguistics* (pp. 170–173). Barcelona, Spain.

Mihalcea, R., Corley, C., & Strapparava, C. (2006). Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the 21st national conference on artificial intelligence*. Boston, MA, USA.

Mihalcea, R., & Tarau, P. (2005). An algorithm for language independent single and multiple document summarization. In *Proceedings of the 2nd international joint conference on natural language processing* (pp. 19–24). Jeju Island, Korea.

Noble, B., & Daniel, J. W. (1988). *Applied linear algebra*. Englewood Cliffs, NJ: Prentice Hall.

Paice, C. D. (1990). Constructing literature abstracts by computer: Techniques and prospects. *Information Processing and Management, 26*(1), 171–186.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2001). BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 311–318). Philadelphia, PA, USA.

Pirolli, P., Pitkow, J., Rao, R. (1996). Silk from a sow's ear: Extracting usable structures from the Web. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 118–125). Vancouver, BC, Canada.

Quillian, M. R. (1968). Semantic memory. In M. R. Minsky (Ed.), *Semantic information processing* (pp. 227–270). Cambridge, MA: The MIT Press.

Radev, D. R., Hovy, E., & McKeown, K. (2002). Introduction to the special issue on summarization. *Computational Linguistics, 28*(4), 399–408.

Radev, D. R., Jing, H., Styś, M., & Tam, D. (2004). Centroid-based summarization of multiple documents. *Information Processing and Management, 40*(6), 919–938.

Reimer, U., & Hahn, U. (1988). Text condensation as knowledge base abstraction. In *Proceedings of the 4th conference on artificial intelligence applications* (pp. 338–344). San Diego, CA, USA.

Salton, G., & McGill, M. J. (1983). *Introduction to modern information retrieval*. New York, NY: McGraw-Hill.

Salton, G., Singhal, A., Mitra, M., & Buckley, C. (1997). Automatic text structuring and summarization. *Information Processing and Management, 33*(2), 193–207.

White, M., Korelsky, T., Cardie, C., Ng, V., Pierce, D., & Wagstaff, K. (2001). Multidocument summarization via information extraction. In *Proceedings of the 1st international conference on human language technology research* (pp. 1–7). San Diego, CA, USA.

Yeh, J.-Y., Ke, H.-R., Yang, W.-P., & Meng, I.-H. (2005). Text summarization using a trainable summarizer and latent semantic analysis. *Information Processing and Management, 41*(1), 75–95.

Zhang, Z., Blair-Goldensohn, S., & Radev, D. R. (2002). Towards CST-enhanced summarization. In *Proceedings of 18th national conference on artificial intelligence* (pp. 439–445). Edmonton, Alberta, Canada.

Zhang, J., Sun, L., & Zhou, Q. (2005). A cue-based hub-authority approach for multi-document text summarization. In *Proceedings of the 2005 IEEE international conference on natural language processing and knowledge engineering* (pp. 642–645). Wuhan, China.

Ziegler, C.-N., & Lausen, G. (2004). Spreading activation models for trust propagation. In *Proceedings of the 2004 IEEE international conference on e-technology, e-commerce and e-service* (pp. 83–97). Taipei, Taiwan.