MWI-Sum: A Multilingual Summarizer Based on Frequent Weighted Itemsets

ELENA BARALIS and LUCA CAGLIERO, Politecnico di Torino
ALESSANDRO FIORI, IRCC: Institute for Cancer Research at Candiolo
PAOLO GARZA, Politecnico di Torino

Multidocument summarization addresses the selection of a compact subset of highly informative sentences, i.e., the summary, from a collection of textual documents. To perform sentence selection, two parallel strategies have been proposed: (a) apply general-purpose techniques relying on data mining or information retrieval techniques, and/or (b) perform advanced linguistic analysis relying on semantics-based models (e.g., ontologies) to capture the actual sentence meaning. Since there is an increasing need for processing documents written in different languages, the attention of the research community has recently focused on summarizers based on strategy (a).

This article presents a novel multilingual summarizer, namely MWI-Sum (Multilingual Weighted Itemset-based Summarizer), that exploits an itemset-based model to summarize collections of documents ranging over the same topic. Unlike previous approaches, it extracts frequent weighted itemsets tailored to the analyzed collection and uses them to drive the sentence selection process. Weighted itemsets represent correlations among multiple highly relevant terms that are neglected by previous approaches. The proposed approach makes minimal use of language-dependent analyses. Thus, it is easily applicable to document collections written in different languages.

Experiments performed on benchmark and real-life collections, English-written and not, demonstrate that the proposed approach performs better than state-of-the-art multilingual document summarizers.

Categories and Subject Descriptors: H.2.8 [**Database Applications**]: Data Mining; I.5.4 [**Applications**]: Text Processing

General Terms: Algorithms

Additional Key Words and Phrases: Multilingual summarization, text mining, frequent weighted itemset mining

ACM Reference Format:
Elena Baralis, Luca Cagliero, Alessandro Fiori, and Paolo Garza. 2015. MWI-Sum: A multilingual summarizer based on frequent weighted itemsets. ACM Trans. Inf. Syst. 34, 1, Article 5 (September 2015), 35 pages.
DOI: http://dx.doi.org/10.1145/2809786

1. INTRODUCTION

During recent years, the growth of the Internet has facilitated the publication of a huge mass of textual documents in electronic form, which represent a potentially powerful source of knowledge. The availability of manageable and readable summaries containing the most relevant information would allow a more effective exploration of increasingly large document collections. For instance, the summary of a collection of

Authors' addresses: E. Baralis, L. Cagliero (corresponding author), and P. Garza, Corso Duca degli Abruzzi, 24 10129 Torino (Italy), Dipartimento di Automatica e Informatica, Politecnico di Torino; emails: name.surname@polito.it; A. Fiori, IRCC: Institute for Cancer Research at Candiolo, Strada Provinciale 142 Km. 3.95 10060 Candiolo (Italy); email: alessandro.fiori@ircc.it.

DOI: http://dx.doi.org/10.1145/2809786

news documents ranging over the same topic may provide a synthetic overview of the most relevant news facets without requiring to access the entire document collection. This article specifically addresses the issue of news document summarization. However, the proposed approach is general. Thus, it can find application in diverse contexts like e-learning and social content summarization.

Many research efforts have been devoted to generating concise summaries by selecting a representative subset of document sentences. To accurately select sentences, two complementary strategies can be adopted: (a) apply general-purpose techniques relying on data mining or information retrieval techniques to effectively analyze the textual data distribution, and/or (b) perform linguistic analyses based on (language-dependent) semantics-based models (e.g., ontologies, lexical databases) to capture the underlying sentence meaning. Summarizers exclusively based on strategy (a) are inherently portable to different languages, whereas most existing approaches relying on strategy (b) have been designed to cope with English-written documents.

Since document collections retrievable from the Web are written in a broad range of different languages, an appealing research issue is the development of summarization systems able to generate concise yet informative document summaries in a multilingual context. Hence, summarization systems should be (i) multidocument, i.e., able to produce a unique summary of a collection of textual documents, and (ii) multilingual, i.e., portable to languages other than English. For this reason, in the last years, the attention of the research community has mainly focused on proposing multi-document summarizers based on strategy (a). Most notable examples are: (i) clustering-based summarizers (e.g., Radev et al. [2004], Wang and Li [2010], and Wang et al. [2011]), (ii) graph-based summarizers (e.g., Zhu et al. [2009], Yang et al. [2011], and Baralis et al. [2013b]), (iii) optimization-based summarizers (e.g., Steinberger et al. [2011], Gillick et al. [2009], and Lin and Bilmes [2011]), and (iv) itemset-based summarizers (e.g., Hynek and Jezek [2003] and Baralis et al. [2011, 2012]). Strategies (i)–(iii) evaluate sentences based on the relative importance of the contained terms or pairs of terms. Conversely, itemset-based summarizers analyze the co-occurrences between multiple document terms (two or more). Hence, their models are potentially more accurate than the other general-purpose ones.

We propose a novel multidocument general-purpose itemset-based summarizer, called MWI-Sum (Multilingual Weighted Itemset-based Summarizer), that allows us to effectively cope with multilingual document collections ranging over the same topic. Unlike any existing itemset-based approach [Hynek and Jezek 2003; Baralis et al. 2012, 2011], MWI-Sum summarization process is driven by frequent weighted itemsets [Wang et al. 2000] and not by traditional (unweighted) itemsets to consider only the correlations between highly relevant terms. Furthermore, the proposed summarizer is easily portable to languages other than English because it requires only a very simple language-dependent analysis. Specifically, MWI-Sum is applicable to document collections written in any language for which a stopword list and, optionally, a stemmer are available. Since stemming and stopword filtering are basic preprocessing steps that are available for a large part of the most widespread languages, MWI-Sum can straightforwardly summarize document collections written in languages other than English.

In the context of textual data, frequent weighted itemsets represent term sets that frequently occur and that consist of highly relevant terms in the analyzed documents. Since traditional itemsets do not discriminate between highly relevant terms and not, adopting weighted itemsets instead of traditional itemsets in document summarization allows us to effectively capture most significant correlations among multiple document terms. Term weights measure term relevance in the analyzed collection. To discriminate between sentences containing relevant information and not, MWI-Sum adopts the following stepwise approach: (i) mapping of the textual documents to a weighted transactional data format, (ii) frequent weighted itemset mining from the preprocessed data, and (iii) selection of the most representative sentences by best covering the previously extracted itemsets. Since sentence selection relies on an itemset-based model, it may also consider a worthy subset of co-occurrences among multiple terms neglected by previous strategies (e.g., Takamura and Okumura [2009] and Conroy et al. [2004, 2011]).

Item weights, which represent term relevance estimates, are generated first and then they are used to drive the extraction of the most relevant itemsets. Weights are assigned using a variant of the established tf-idf (term frequency-inverse document frequency) statistics [Tan et al. 2002], which hereafter we will denote as tf-df (term frequency-document frequency). Specifically, to tailor the tf-idf interestingness measure to the analysis of homogeneous document collections, global term document frequency positively contributes to term evaluation, similar to local term frequency.

We validated the effectiveness of our approach on both multilingual [Text Analysis Conference 2011] and English-written [Document Understanding Conference 2004] benchmark document collections as well as on a set of real-life multilingual news articles that were published by the most renowned newspapers. The experiments demonstrate that MWI-Sum is, on average, more effective than all the multilingual summarizers presented to the MultiLing Pilot of TAC'11 contest [Giannakopoulos et al. 2011] as well as than state-of-the-art summarizers (itemset-based and not) on both benchmark and real multilingual collections. Furthermore, considering item weights during the itemset extraction process is particularly useful for summarizing documents written in languages other than English.

The article is organized as follows. Section 2 compares our work with previous approaches. Section 3 describes the main steps of the MWI-Sum summarizer, while Section 4 experimentally evaluates its performance on different document collections. Finally, Section 5 draws conclusions and presents future work.

2. RELATED WORK

Document summarization aims at generating concise summaries that describe the content of one or more textual documents. Depending on the type of generated summaries, two main approaches to text summarization have been proposed. Sentence-based approaches entail partitioning the document(s) into sentences and selecting the most informative ones to include in the summary [Carenini et al. 2007; Mittal et al. 2000; Wang and Li 2010; Wang et al. 2011]. Conversely, keyword-based approaches focus on detecting salient keywords to summarize the document content using either co-occurrence measures [Lin and Hovy 2003] or latent semantic analysis [Dredze et al. 2008]. The summarization system proposed in this article relies on a sentence-based approach.

To effectively perform sentence selection different strategies have been adopted: (i) clustering, (ii) graph ranking, (iii) optimization strategies, and (iv) frequent itemset mining. The summarizer presented in this article relies on an itemset-based approach. However, for the sake of completeness, a detailed comparison with all the main existing strategies is given below.

Clustering-based approaches (e.g., Radev et al. [2004], Wang and Li [2010], and Wang et al. [2011]) exploit clustering algorithms to address document summarization. For example, in Wang et al. [2011], clusters represent groups of sentences from which the best representatives (e.g., the centroids or the medoids) are selected. In contrast, the MEAD text summarizer [Radev et al. 2004] clusters documents instead of single sentences and it evaluates the corresponding cluster centroids. In this context, a centroid is a pseudo-document that consists of sentences selected by evaluating the tf-idf values [Lin and Hovy 2003] of the corresponding terms. In Wang and Li [2010], an incremental clustering algorithm similar to those previously proposed in Guha et al. [2003] is exploited to address the issue of dynamic summary update. Once a set of documents is added/removed from the analyzed collection, the previously generated summary is updated without the need for recalculating the whole clustering model. Clustering algorithms are particularly suitable for partitioning documents/sentences ranging over different topics. Conversely, this article specifically addresses the problem of summarizing documents ranging over the same topic.

Graph-based approaches (e.g., Radev [2004], Thakkar et al. [2010], Wan and Yang [2006], Zhu et al. [2009], Yang et al. [2011], and Baralis et al. [2013b]) construct a graph whose nodes represent document sentences, while edges connect pairs of nodes and are weighted by pairwise node similarity measures. To reduce the computational complexity, edges are early pruned by enforcing a minimum similarity threshold. Then, popular indexing strategies (e.g., PageRank [Brin and Page 1998]) are used to extract the most authoritative sentences. To further improve the summarization performance, Zhu et al. [2009] and Yang et al. [2011] propose a semantics-based approach that combines the knowledge provided by the user-generated content (e.g., document tags and contextual information) with graph-based models. More recently, the GraphSum summarizer [Baralis et al. 2013b] has been presented. It first exploits association rules between pairs of terms to generate the graph, then ranks sentences by using the HITS algorithm [Kleinberg 1999]. While graph-based approaches consider only pairwise correlations between terms, our itemset-based approach considers higher-order dependences among terms as well.

Optimization strategies have also been applied to accomplish the summarization task. For example, Steinberger et al. [2011] perform Singular Value Decomposition (SVD) to extract salient sentences in a multilingual context, while in Gillick et al. [2009, 2008] and Lin and Bilmes [2011] Integer Linear Programming and submodular function optimization techniques are exploited, respectively. In Takamura and Okumura [2009], Filatova [2004], and Wang et al. [2013] the sentence selection process is formalized as a min-max optimization problem and it is tackled by means of combinatorial optimization strategies. A similar approach has been adopted in Wang et al. [2013] to summarize differences among document collections. In Conroy et al. [2004, 2011], different versions of the CLASSY summarizer have been presented to the generic multidocument DUC'04 and the multilingual TAC'11 summarization contests, respectively. They perform document summarization by combining optimization with hidden Markov models. While the approaches presented in Conroy et al. [2004], Takamura and Okumura [2009], Filatova [2004], Wang et al. [2013], and Gillick et al. [2008, 2009] are specifically tailored to English-written documents, the summarizers in Steinberger et al. [2011] and Conroy et al. [2011] have been designed to cope with multilingual documents. Unlike Steinberger et al. [2011] and Conroy et al. [2011], the newly proposed MWI-Sum summarizer exploits frequent weighted itemsets to identify salient correlations among document terms. The proposed approach appears to be more effective than [Steinberger et al. 2011; Conroy et al. 2011] on benchmark multilingual documents and it performs as well as state-of-the-art approaches on English-written documents.

Frequent itemset mining from transactional datasets is a well-known data mining problem [Han et al. 2007]. Some attempts to exploit this technique in document summarization have already been made. In this context, itemsets represent co-occurrences among document terms. To treat items differently within each transaction based on their relevance/interest, the concept of weighted itemset has also been introduced [Wang et al. 2000; Sun and Bai 2008]. Item weights can be known in advance [Wang et al. 2000] or inferred by using indexing strategies [Sun and Bai 2008]. Since the set of frequent itemsets is inherently redundant, research efforts have also been devoted to selecting a succinct subset of itemsets [Jaroszewicz and Simovici 2004; Tatti 2010; Mampaey et al. 2011]. Unlike [Jaroszewicz and Simovici 2004; Tatti 2010; Mampaey et al. 2011], this paper focuses on selecting document sentences instead of itemsets. To the best of our knowledge, the previous works presented in Hynek and Jezek [2003] and Baralis et al. [2011, 2012] are preliminary attempts to exploit itemsets in document summarizations. In Hynek and Jezek [2003], itemsets (i.e., sets of terms) are extracted from a transactional dataset, where each transaction corresponds to a distinct document and it consists of its corresponding set of (not repeated) terms. Itemsets are then used to compose summary sentences, i.e., the summary sentences are "abstracted" from the itemset-based model. The approach proposed in this article significantly differs from Hynek and Jezek [2003] because summaries are not "abstracted" from the itemsets, but they are generated by combining existing sentences. More recently, Baralis et al. [2011, 2012] consider each document sentence as a distinct transaction and they exploit an established entropy-based strategy to generate compact itemset-based models. The itemsets are then used to summarize English-written documents. The summarizer presented in this article differs, to a great extent, from those proposed in Baralis et al. [2011, 2012]. The major differences can be summarized as follows: (i) the use of weighted itemsets instead of traditional (unweighted) ones to push item relevance deep into the mining process; (ii) the use of a new relevance score, i.e., the term frequency-document frequency (tf-df), which is more appropriate than tf-idf and entropy-based statistics to cope with homogeneous document collections; and (iii) significantly improved performance on benchmark and real multilingual documents. As discussed in Section 4, item weights appear to be particularly useful for improving the effectiveness of the summarization process on documents written in languages other than English. A somewhat different approach is presented in Gross et al. [2014]. The authors first consider pairwise associations between document terms to weigh the importance of each document sentence. Then, they greedily select the subset of the most important sentences. This approach neglects higher-order associations between terms (e.g., combinations of three or more terms). Furthermore, since no early sentence pruning is performed yet, pretty uninformative sentences may be picked out as well.

Parallel efforts have been devoted to enriching the sentence selection process with linguistic or semantics-based analyses. For example, ontologies have been exploited to identify (i) the concepts that are most pertinent to a user-specified query [Kogilavani and Balasubramanie 2009; Ping and M. 2006; Baralis et al. 2013a]; (ii) the context in which summaries are generated in different application domains (e.g., the context-aware mobile domain [Fortes et al. 2006], the business domain [Wu and Liu 2003], the disaster management domain [Li et al. 2010]); and (iii) the text argumentative structure [Pourvali and Abadeh 2012; Atkinson and Munoz 2013; Hennig et al. 2008]. Notable examples of summarizers based on linguistic analyses are those relying on lexical chains [Barzilay and Elhadad 1997; Morris and Hirst 1991; Wu et al. 2010]. A lexical chain is a sequence of related words in writing, spanning short (adjacent words or sentences) or long distances (entire text). A chain is independent of the grammatical structure of the text and in effect it is a list of words that captures a portion of the cohesive structure of the text. However, lexical chains are commonly derived from ontologies or lexical databases (e.g., Wordnet [Database 2012]). Conversely, itemsets are mined directly from the raw data without the need for semantics-based models. Therefore, while the effectiveness of lexical chains in text summarization depends on the accuracy of the language-dependent model, the use of itemsets in text summarization is independent of the language in which the input documents are written. For this reason, we exploit itemsets rather than lexical chains to address multilingual document summarization.

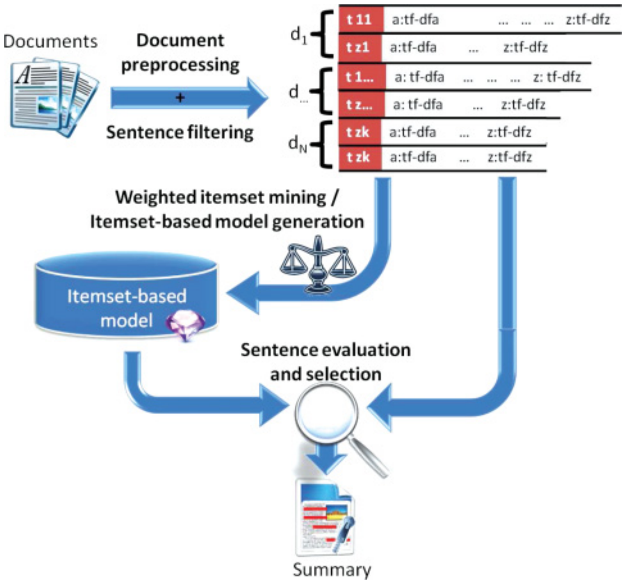

Documents
Document preprocessing
+
Sentence filtering
d1
t 11 a:tf-dfa … … … z:tf-dfz
t z1 a:tf-dfa … z:tf-dfz
d…
t 1… a: tf-dfa … … … z: tf-dfz
t z… a: tf-dfa … z:tf-dfz
dN
t zk a:tf-dfa … z:tf-dfz
t zk a:tf-dfa … z:tf-dfz
Weighted itemset mining /
Itemset-based model generation
Itemset-based model
Sentence evaluation and selection
Summary

Fig. 1. The MWI-Sum summarizer.

3. THE SUMMARIZATION APPROACH

M W I-Sum (Multilingual Weighted Itemset-based Summarizer) is a novel multidocument and multilingual summarizer that is designed for document collections ranging over the same topic and written in different languages. Figure 1 outlines its main steps, which are summarized below.

—**Document preprocessing.** To suit the input document collection to the subsequent mining process, stopword and stemming algorithms are applied as the only (basic) language-dependent analysis. Furthermore, based on the considered language, stemming can be selectively enabled/disabled to tune system performance. Finally, documents are transformed in a weighted transactional data format by mapping each term to a *relevance score*.

—**Sentence filtering.** To filter out the sentences of marginal interest based on their relative position in the document, sentences placed at the end of each document are early pruned.

—**Frequent weighted itemset mining and itemset-based model generation.** The subset of the frequent weighted itemsets, called *itemset-based model* throughout the article, is generated from the weighted transactional representation of the given document collection. Frequent weighted itemsets are recurrent co-occurrences among sets of highly relevant terms contained in the analyzed documents.

—**Sentence evaluation and selection.** To identify the most representative and not redundant sentences to include in the summary, sentences that best cover the previously extracted itemset-based model are selected by means of a greedy strategy.

A more thorough description of each step is reported in the following sections.

3.1. Document Preprocessing

Let us consider a textual document collection $D = \{d_1, \ldots, d_N\}$ that consists of N documents. This block focuses on preparing the raw collection D for the subsequent mining steps. To this aim, it entails the following steps: (i) text processing and (ii) term relevance evaluation.

3.1.1. Text Processing. Language-dependent processing takes place only in this step. A stopword elimination preprocessing step filters out the words that usually have little lexical content, because the itemset mining process fails to distinguish them from the other words. Specifically, the filter searches for a matching between the text words and the words that are contained in a (language-dependent) stopword corpus. To this purpose, we adopted the Natural Language Toolkit (NLTK) stopword corpus [Loper and Bird 2002]. Furthermore, a stemming algorithm is applied to reduce document words to their base or root form (i.e., the stem). More specifically, the Snowball stemmer [Bird et al. 2009] is used for the English language, while the Lucene stemmer [McCandless et al. 2010] is exploited for managing the other languages. Stemming and stopword elimination are basic language-dependent algorithms that are available for a large part of the most widespread languages. MWI-Sum only integrates, as language-dependent processing steps, the two aforementioned algorithms. Hence, it is straightforwardly portable to languages other than English.

Based on the considered language, stemming can be selectively enabled/disabled to tune system performance. For example, as discussed in Section 4, on documents written in non-Latin languages (e.g., Hindi), enabling the Lucene stemmer prior to summarization yields a worse summarization performance. Conversely, on most Latin languages (e.g., English), enabling the stemmer yields significant performance improvements.

The result of the stemming and stopword elimination steps is a bag-of-word (BOW) representation [Tan et al. 2002] of the document collection D, in which each processed document $d_k \in D$ consists of a set of sentences $S_k = \{s_{1k}, \ldots, s_{zk}\}$, where each sentence contains an unordered set of word stems, called *terms* throughout the article. Each term may occur more than once within each sentence.

3.1.2. Term Relevance Evaluation. Terms contained in each sentence have not the same importance in the analyzed collection. We assign to each sentence term a score that denotes its importance in the whole collection. To this aim, we introduce a variant of the established tf-idf (term frequency-inverse document frequency) statistics [Lin and Hovy 2003] named tf-df (term frequency-document frequency). To introduce the tf-df statistics, we preliminarily recall the definition of tf-idf.

The tf-idf Statistics. The term frequency-inverse document frequency (tf-idf) evaluator [Lin and Hovy 2003] is an established and widely used statistics that is intended to reflect how important a term is in a document of a collection or corpus [Tan et al. 2002]. Tf-idf is thus defined as follows:

$$ti_{ik} = \frac{n_{ik}}{|d_k|} \cdot \log \frac{|D|}{|\{d_k \in D \ : \ w_i \in d_k\}|}, \tag{1}$$

where n_{ik} is the number of occurrences of the i-th term w_i in the k-th document d_k, D is the document collection, $|d_k|$ is the number of terms that are contained in the k-th document d_k, and $\frac{|D|}{|\{d_k \in D \ : \ w_i \in d_k\}|}$ represents the inverse document frequency of the term w_i in the whole collection. The logarithm of the inverse document frequency is minimal when the inverse document frequency is equal to 1 (i.e., a term occurs in every document of the collection) and thus the corresponding td-idf value reduces to zero.

The key idea behind the tf-idf statistics is that terms appearing frequently in a few documents (i.e., high local term frequency), but rarely in the whole collection (i.e., low document frequency), are the most effective ones in discriminating among sentences in a collection. Although the tf-idf statistics is a reliable term relevance estimate for document summarization purposes when dealing with heterogeneous textual

Table I. Running Example D

Document	Sentence ID	Sentence
d_1	1	Beautiful, World
	2	World, Century, Change
d_2	3	World, Fantasy
	4	World, Fantasy, Change
d_3	5	Internet, Century
	6	Era, Internet

Table II. Tf-idf and td-df Statistics

	d_1		d_2		d_3	
	Tf-df	Tf-idf	Tf-df	Tf-idf	Tf-df	Tf-idf
Beauty	0.067	0.095	0	0	0	0
World	0.267	0.070	0.267	0.070	0	0
Century	0.134	0.035	0	0	0.167	0.044
Change	0.134	0.035	0.134	0.035	0	0
Fantasy	0	0	0.134	0.191	0	0
Internet	0	0	0	0	0.167	0.239
Era	0	0	0	0	0.084	0.119

documents [Lin and Hovy 2003; Radev et al. 2004; Wang et al. 2011], it shows some limitations when coping with documents ranging over the same topic [Lin and Hovy 2003]. More specifically, the most relevant terms that appear in a homogeneous collection (e.g., a collection of news articles that range over the same subject) are likely to occur frequently both in each single document and in the whole collection. Hence, we adopt a variant of the tf-idf statistics, named tf-df (term frequency-document frequency), that provides a more reliable term relevance estimate than tf-idf when coping with homogeneous documents.

The tf-df Statistics. The term *frequency document frequency (tf-df) evaluator* is a newly proposed variant of the tf-idf statistics, which is defined as follows:

$$td_{ik} = \frac{n_{ik}}{|d_k|} \cdot \frac{|\{d_k \in D \ : \ w_i \in d_k\}|}{|D|}. \tag{2}$$

Unlike tf-idf, in the tf-df statistics, the document frequency of a term positively contributes to the term evaluation. The tf-df value reduces to zero when a term never occurs in the analyzed collection, while the document frequency contribution becomes maximal when a term occurs in all documents. In other words, the more a term is frequent in the whole collection, the more it is important in the analyzed collection. This assumption is reliable when coping with collections of documents ranging over the same subject.

To clarify the difference between tf-df and tf-idf statistics, let us consider the document collection reported in Table I, which will be used as a running example throughout the section. The collection contains three documents, each one composed of two sentences. For each sentence, the corresponding terms, which were obtained after the stemming and stopword elimination steps, are reported. For example, the first sentence of document d_1 contains the terms *Beauty* and *World*. Table II reports the tf-idf and tf-df values that are associated with the terms contained in the collection. Terms that occur frequently in a single document but rarely in the whole collection (e.g., *Fantasy*) are characterized by high tf-idf value and low tf-df value, whereas terms that occur frequently both locally (i.e., within each document) and globally in the collection

Table III. Matching Weights of Itemsets {Century} and {World, Change} in D

Document	TID	Transaction	Itemset {Century} weight	Itemset {World, Change} weight
d_1	1	$\{\langle Beauty, 0.067\rangle, \langle World, 0.267\rangle\}$	0	0
	2	$\{\langle World, 0.267\rangle, \langle Century, 0.134\rangle, \langle Change, 0.134\rangle\}$	0.134	0.134
d_2	3	$\{\langle World, 0.267\rangle, \langle Fantasy, 0.134\rangle\}$	0	0
	4	$\{\langle World, 0.267\rangle, \langle Fantasy, 0.134\rangle, \langle Change, 0.134\rangle\}$	0	0.134
d_3	5	$\{\langle Internet, 0.167\rangle, \langle Century, 0.167\rangle\}$	0.167	0
	6	$\{\langle Era, 0.084\rangle, \langle Internet, 0.167\rangle\}$	0	0

(e.g., *World*) are characterized by low tf-idf and high tf-df score. An experimental evaluation of the effectiveness of the tf-df score in the context of homogeneous document summarization is given in Section 4.6.

3.2. Sentence Filtering

Document collections frequently contain redundant information. Early pruning less meaningful or redundant parts may improve the effectiveness and efficiency of the summarization process. For this reason, we apply sentence filtering prior to document summarization. Specifically, similar to state-of-the-art sentence-based approaches (e.g., Conroy et al. [2004, 2011] and Gillick et al. [2008, 2009]) the MWI-Sum summarizer considers the sentence position to perform early sentence pruning.

For each document in the collection MWI-Sum considers, for the subsequent analysis, only the top-K sentences of the document, i.e., the first-K sentences of each document (where K is an input parameter provided by the analyst), while it filters out all the remaining ones because they are less likely to be relevant for summarization purposes. As discussed in Conroy et al. [2004], the reliability of this assumption depends on the nature of the analyzed documents. For example, when coping with news documents, top-placed sentences are most likely to summarize all key concepts. In the contexts in which the above assumption may be inappropriate, the users can disable the sentence early pruning option. An experimental analysis of the impact of parameter K on summarization performance is reported in Section 4.

3.3. Itemset-Based Model Generation

Discovering relevant co-occurrences among multiple terms in the analyzed collection is a key point to generate high-quality summaries. To this aim, MWI-Sum generates a model that consists of frequent weighted itemsets, i.e., sets of terms that frequently occur together and that are characterized by relatively high relevance in the source collection.

Before the itemset mining process, the preprocessed document collection is mapped to a transactional data format, where transactions correspond to document sentences, while items correspond to sentence terms. Each item is weighted by a score indicating the relevance of the term within the corresponding sentence and document. Recall the running example reported in Table I. Column 3 of Table III reports the transactional representation of the document collection, which hereafter we will denote as *transactional dataset*. The dataset contains six transactions, one for each document sentence.

A weighted transaction tr_{jk} in the dataset corresponds to sentence s_{jk}, i.e., the j-th sentence of the k-th document in the collection. The transaction contains a set of pairs $\langle w_q, td_{qk}\rangle$, where w_q is a distinct term in sentence s_{jk} and td_{qk} is the tf-df value of term w_q in s_{jk}. For example, the transaction with TID 1 in Table III contains two weighted items, because the first sentence of document d_1 in Table I contains terms *Beauty* and

Table IV. Weighted Support Evaluation for the Itemsets {Century} and {World, Change} in *D*

Document	TID	Transaction	Normalization Factor	Itemset {Century} matching weight	Itemset {World, Change} matching weight
d_1	1	$\{\langle \mathit{Beauty}, 0.067\rangle, \langle \mathit{World}, 0.267\rangle\}$	0.267	0	0
	2	$\{\langle \mathit{World}, 0.267\rangle, \langle \mathit{Century}, 0.134\rangle, \langle \mathit{Change}, 0.134\rangle\}$	0.267	0.134	0.134
d_2	3	$\{\langle \mathit{World}, 0.267\rangle, \langle \mathit{Fantasy}, 0.134\rangle\}$	0.267	0	0
	4	$\{\langle \mathit{World}, 0.267\rangle, \langle \mathit{Fantasy}, 0.134\rangle, \langle \mathit{Change}, 0.134\rangle\}$	0.267	0	0.134
d_3	5	$\{\langle \mathit{Internet}, 0.167\rangle, \langle \mathit{Century}, 0.167\rangle\}$	0.167	0.167	0
	6	$\{\langle \mathit{Era}, 0.084\rangle, \langle \mathit{Internet}, 0.167\rangle\}$	0.167	0	0
Total			**1.402**	**0.301**	**0.268**
$\mathcal{W}sup$				**21.4%**	**19.1%**

World. Term occurrences are weighted by the corresponding tf-df values. For instance, the weight of term *Beauty* is 0.067.

In our context, an itemset I of length k (i.e., a k-itemset) is a set of k distinct terms. The support of an itemset in a transactional dataset T is its frequency of occurrence in T. Since items are characterized by different relevance scores in different transactions, the frequency count of the itemset in a dataset may be weighted by the corresponding transaction weights [Wang et al. 2000]. In such a way, itemsets containing high-relevance scores have, on average, higher support values. More specifically, given an itemset I, we define the *matching weight* of I in a weighted transaction tr_{jk} as its least item weight (i.e., the lowest tf-df value).

Table III reports the matching weights of {*Century*} and {*World, Change*} with respect to the transactions contained in the running example collection D. Weights were obtained from the tf-df values reported in Table II. The matching weight of the same weighted itemset may change from one transaction to another, because itemsets matching sentences that belong to different documents may have different weights in different sentences (e.g., the itemset {*Century*} has matching weights 0.134 and 0.167 with respect to the sentences with TID 2 and 5, respectively).

Similar to Cagliero and Garza [2014], the support of an itemset I in a weighted transactional dataset T is defined as a linear combination of the matching weights with respect to each transaction in T. Its expression follows.

Definition 3.1 (*Weighted Support*). Let I be a weighted itemset and T the corresponding weighted transactional dataset. The weighted support of I in T is defined as follows.

$$\mathcal{W}sup(I, T) = \frac{\sum_{tr_{jk} \in T} \mathcal{W}(I, tr_{jk})}{\sum_{tr_{jk} \in T} \max_{i|w_i \in tr_{jk}} td_{ik}} = \frac{\mathcal{W}(I, T)}{\sum_{tr_{jk} \in T} \max_{i|w_i \in tr_{jk}} td_{ik}}$$

While the numerator $\mathcal{W}(I, T)$, hereafter denoted as *absolute weighted support*, is the summation of all the itemset matching weights for every transaction in T, the denominator is a normalization factor that is computed by summing the maximum item weights in each transaction of T. For example, as shown in Table IV, the weighted supports of {*Century*} and {*World, Change*} are $\frac{0.301}{1.402} = 21.4\%$ and $\frac{0.268}{1.402} = 19.1\%$, respectively.

Weighted itemsets whose support exceeds a minimum threshold are said to be *frequent*. The weighted support measure is characterized by the following two properties.

Property 3.1 [Equivalence between Weighted Support and Support]. *Let D be a document collection whose corresponding weighted transactional dataset T exclusively* *contains items with weight 1. The weighted support value of an itemset I in T corresponds to its (traditional) support value in D, i.e., $\mathcal{W}sup(I, T) = support(I, D)$.*

PROOF. Since every item in T has a weight equal to 1, if $tr_{jk} \in T$ contains all the items in I, then $W(I, tr_{jk}) = 1$; otherwise, $W(I, tr_{jk}) = 0$. Hence, from Definition 3.1, the weighted support expression is given by (i) as numerator, the number of weighted transactions that are matched by I and (ii) as denominator, the dataset cardinality $|T| = |D|$. Hence, it follows that $\mathcal{W}sup(I, T) = \text{support}(I, D)$. □

Itemset mining is typically driven by a minimum support constraint, i.e., itemsets whose support is above a given threshold are extracted. Since the minimum support constraint is antimonotone, part of the search space may be pruned early. The antimonotonicity property holds for the minimum weighted support constraint as well.

PROPERTY 3.2 [ANTI-MONOTONICITY OF THE WEIGHTED SUPPORT CONSTRAINT]. *Let T be a weighted transactional dataset. Let $\preceq$ be a precedence relation that is defined on a pair of weighted itemsets X and Y, such that $X \preceq Y$ holds if and only if $X \subseteq Y$. Let ξ be a nonnegative number. The minimum weighted support constraint $\mathcal{W}sup(X, T) \geq \xi$ is antimonotone with respect to $\preceq$.*

PROOF. Since $X \preceq Y$ then the set of transactions matched by Y is a subset of the transactions matched by X. Furthermore, given an arbitrary transaction tr_{jk} matched by both X and Y, it trivially follows that the matching weight of Y with respect to tr_{jk} is at most equal to the matching weight of X with respect to tr_{jk}, i.e., $\mathcal{W}(X, tr_{jk}) \geq \mathcal{W}(Y, tr_{jk})$. Hence, the following inequality holds.

$$\mathcal{W}sup(X, T) = \frac{\sum_{tr_{jk} \in T} \mathcal{W}(X, tr_{jk})}{\sum_{tr_{jk} \in T} \max_{i|w_i \in tr_{jk}}(td_{ik})} \geq \frac{\sum_{tr_{jk} \in T} \mathcal{W}(Y, tr_{jk})}{\sum_{tr_{jk} \in T} \max_{i|w_i \in tr_{jk}}(td_{ik})} = \mathcal{W}sup(Y, T)$$

It follows that the minimum weighted support constraint is antimonotone with respect to the precedence relation $\preceq$. □

The above property will be exploited by the MWI-Sum algorithm to effectively generate an itemset-based model, which consists of all the frequent weighted itemsets extracted from the document collection by enforcing a minimum weighted support threshold $\mathcal{W}minsup$, i.e., all weighted itemsets whose weighted support is above or equal to $\mathcal{W}minsup$.

3.4. Frequent Weighted Itemset Mining

MWI-Sum exploits a projection-based algorithm [Grahne and Zhu 2003] to perform the extraction of frequent weighted itemsets. To prune the search space early, MWI-Sum exploits the antimonotonicity property of the minimum weighted support constraint (see Property 3.2).

Since traditional projection-based algorithms are not directly suitable for coping with weighted transactional datasets, we exploited a mining strategy similar to the one previously proposed in Cagliero and Garza [2014] in the context of infrequent itemset mining. Specifically, the adopted itemset miner exploits (i) an equivalence property, which makes the transactional dataset suitable for being stored in an FP-tree structure [Han et al. 2000] and (ii) a slightly modified FP-tree creation and population procedure, to efficiently retrieve the weighted support of the candidate itemsets. Itemset extraction is performed on an equally weighted transactional representation of the analyzed dataset, in which each transaction et_p exclusively contains equally weighted items with weight tde_p. The equivalent dataset transactions are stored in a slightly modified FP-tree index, in which weighted support values are stored rather than traditional support values. Unlike the traditional FP-Growth-like tree population, the insertion of a new transaction et_p triggers the update of the weight of the corresponding FP-tree nodes by tde_p instead of by 1.

In the following, the transaction equivalence property is formally stated.

3.4.1. Transaction Equivalence. The weighted transaction equivalence associates with a weighted transactional dataset T composed of transactions that include arbitrarily weighted items an equally weighted transaction representation ET in which each transaction is exclusively composed of equally weighted items. Specifically, to each weighted transaction in T a corresponding set of equally weighted transactions in ET is associated. The proposed transformation aims at representing the original dataset by means of an FP-tree [Han et al. 2000] to perform an FP-growth-like itemset mining.

Definition 3.2 (*Equally Weighted Transactional Representation*). Let $tr = \{\langle w_1, td_1\rangle, \langle w_2, td_2\rangle, \ldots, \langle w_m, td_m\rangle\}$ be a weighted transaction. Let $k \leq m$ be the number of distinct weights associated with terms in tr and $\overline{W} = \{td_{1^*}, \ldots, td_{k^*}\}$ the enumeration of the distinct weights in increasing order. The equally weighted transactional representation $ET_{tr} = \{et_1, \ldots, et_k\}$ associated with tr consists of k weighted transactions, where each transaction et_p ($p \in [1, k]$) contains items with weight tde_p and is defined as follows:

$$et_p = \{\langle w_j, tde_p\rangle \mid \langle w_j, td_j\rangle \in tr \wedge \ td_j \geq td_{p^*}\},$$

where

—td_{p^*} is the p-th element in $\overline{W}$

—$tde_p = \begin{cases} td_{1^*} & \text{if } p = 1, \\ td_{p^*} - td_{p-1^*} & \text{otherwise} \end{cases}$

The equally weighted representation ET of a weighted transactional dataset T is the union of all the equally weighted representations ET_{tr} associated with each weighted transaction in T.

Recalling the running example, the equally weighted representation of transaction $tr = \{\langle World, 0.267\rangle, \langle Century, 0.134\rangle, \langle Change, 0.134\rangle\}$ in T (see Table III) consists of the following transactions:

(1) $et_1 = \{\langle World, 0.134\rangle, \langle Century, 0.134\rangle, \langle Change, 0.134\rangle\}$
(2) $et_2 = \{\langle World, 0.133\rangle\}$

et_1 contains all the items in tr, each one weighted by the least term weight (0.0134), while et_2 contains term *World* solely, because it represents the not fully covered terms. Its weight is set to 0.133 (= 0.267−0.134).

The support of a weighted itemset in a weighted transactional dataset T is equal to the one evaluated on its equally weighted representation. We denote this property as equivalence property. Its formal definition follows.

PROPERTY 3.3 [EQUIVALENCE PROPERTY]. *Let T be a weighted transactional dataset and ET its equally weighted representation. The weighted supports of an arbitrary weighted itemset I in T and ET are equal.*

The proof of the above property is reported in Appendix.

3.5. Sentence Evaluation and Selection

MWI-Sum generates the summary of a document collection by exploiting the itemset-based model to extract the most relevant sentences. Sentences are evaluated by considering (a) their coverage of the itemset-based model and (b) their relevance, measured by a relevance score based on the tf-df statistics associated with each sentence term.

The selection of the sentences with the best model coverage and relevance score can be modeled as a set covering problem. MWI-Sum addresses this problem by means of a greedy heuristics.

The relevance score of a sentence s_{jk} in a document collection D measures the significance of a sentence in terms of the tf-df values of its terms in D. It is defined as the sum of the tf-df values of its terms.

Definition 3.3 (*Sentence Relevance Score*). Let s_{jk} be a sentence and t_{jk} the corresponding transaction. The sentence relevance score of sentence s_{jk} is given by

$$SR(s_{jk}) = \frac{\sum_{i \ |w_i \in t_{jk}} td_{ik}}{|t_{jk}|}$$

Sentence coverage measures the pertinence of a sentence to the generated itemset-based model.

Definition 3.4 (*Sentence Coverage*). Let M be the itemset-based model of a document collection D. The coverage of a sentence s_{jk} with respect to M is defined as the number of itemsets $I \in M$ matching s_{jk}.

To measure sentence coverage, we associate with each sentence $s_{jk} \in D$ a binary vector, hereafter denoted as *sentence coverage vector*, $SC_{jk} = \{sc_1, \ldots, sc_{ms}\}$, where ms is the number of itemsets belonging to model M. Each vector element sc_i is 1 if itemset I_i matches tr_{jk}, and 0 otherwise. Hence, the coverage of a sentence s_{jk} with respect to the itemset-based model M is computed as the number of ones that occur in the corresponding coverage vector SC_{jk}.

The problem of selecting the most representative sentences to generate a document collection summary can be formulated as a set covering problem. More specifically, the set covering optimization problem focuses on selecting the minimal set of l sentences with maximal score, whose logic OR of the corresponding coverage vectors $SC^* = SC_1 \vee \ldots \vee SC_l$ generates a binary vector with the maximum number of 1s. The SC^* vector will be denoted as *summary coverage vector* throughout this section.

Since the set covering optimization problem is NP-hard, we tackled it by adopting a greedy strategy for sentence selection. The adopted heuristics considers sentence model coverage as the most discriminative feature. Hence, sentences that cover the maximum number of itemsets belonging to the itemset-based model are selected first. On equal terms, the sentence with maximal coverage that is characterized by the highest relevance score is preferred. Note that by maximizing weighted itemset coverage, we select the sentences that contain the largest number of frequent term combinations. Unlike statistics-based term evaluators, this term co-occurrences may involve combinations of more than two terms. Hence, they may represent high-order dependences among textual data.

The pseudocode of the greedy approach is reported in Algorithm 1. The algorithm identifies, at each step, the sentence s_{jk} with the best complementary vector SC_{jk} with respect to the current summary coverage vector SC^*, i.e., the sentence s_{jk} that covers the maximum number of itemsets not already covered by any sentence in the current summary. Algorithm 1 takes as input the set of sentences S, the set of sentence coverage vectors SC, and the set of sentence relevance scores SR. It produces a summary $\mathcal{SU}$ that includes the minimal subset of sentences that best covers the itemset-based model. The first step is variable initialization (lines 1 and 2). The summary is initialized as the empty set and the summary coverage vector is set to a binary vector of all zeros (i.e., no itemset is covered yet). Next, the best sentences to include in the summary are iteratively selected (lines 3–13). At each iteration the sentence(s) with maximum coverage, i.e., the one(s) whose sentence coverage vector contains the maximum number

ALGORITHM 1: Greedy sentence selection

```
Require: set of sentences $S$
Require: set of sentence coverage vectors $SC$
Require: set of sentence relevance scores $SR$
Ensure: summary $\mathcal{SU}$
 1: $\mathcal{SU} = \emptyset$
 2: $SC^* =$ set_to_all_zeros() /*initialize the summary coverage vector with only zeros */
    /* Cycle until $SC^*$ contains only 1s (i.e., until the generated summary covers all the itemsets
    of the model) */
 3: while $SC^*$ contains at least one zero do
 4:   MaxOnesSentences = max_ones_sentences($S$, $SC$) /* Select the sentences with the highest
      number of ones */
 5:   if MaxOnesSentences is not empty then
 6:     $s_{\text{best}} = \arg\max_{s_j \in \text{MaxOnesSentences}} SR(s_j)$ /* Select the sentence with maximum relevance
        score among the ones in MaxOnesSentences */
 7:     $\mathcal{SU} = \mathcal{SU} \cup s_{\text{best}}$ /* Add the best sentence to the summary */
        /* Update the summary coverage vector $SC^*$. $SC(s_{best}) \in SC$ is the sentence coverage
        vectorassociated with the best sentence $s_{best}$ */
 8:     $SC^* = SC^*$ OR $SC(s_{\text{best}})$/* Set the bits associated with the itemsets covered by $s_{best}$ to one
        */
        /* Update the sentence coverage vectors in SC */
 9:     for all $SC_i$ in $SC$ do
10:       $SC_i = SC_i$ AND $\overline{SC^*}$ /* Set the bits of $SC_i$ associated with the itemsets already
          covered by the summary to zero */
11:     end for
12:   end if
13: end while
14: return $\mathcal{SU}$
```

of ones, is selected (line 4). In case of ties, the sentence with maximum relevance score (see Definition 3.3) is preferred (line 6). On equal terms, the sentence that ranked first in alphabetical order is chosen. Finally, the selected sentence s_{best} is included in the summary $\mathcal{SU}$ (line 7) and the sentence coverage vectors are updated (lines 8–11). The update step excludes from the set of considered itemsets the ones already covered by the summary. The procedure iterates until the summary coverage vector contains only ones, i.e., until the itemset-based model is fully covered by the summary (line 3). Since the frequent itemsets that compose the model M are mined from the transactional weighted representation of D, it follows that, when $\mathcal{W}minsup \neq 0$ each frequent itemset I in M matches at least one document sentence s in D, i.e., s covers I. Hence, the following property is fulfilled.

PROPERTY 3.4. *Let D be a document collection and M the* MWI-S*um itemset-based model that is built on D by enforcing a minimum weighted support threshold $\mathcal{W}minsup > 0$. The summary $\mathcal{SU}$ generated by* MWI-S*um fully covers M.*

The proof of the above property is reported in Appendix.

The experimental results reported in Section 4.6 show that the use of the greedy sentence selection algorithm is more effective and efficient than using a branch-and-bound algorithm for summarization purposes.

4. EXPERIMENTAL RESULTS

We performed a large set of experiments to (i) compare the performance of MWI-Sum and other state-of-the-art summarizers on multilingual and English-written benchmark datasets (Sections 4.2 and 4.3), (ii) experimentally evaluate the effectiveness of the proposed summarizer on real-life news articles (Section 4.4), (iii) analyze the effect of item weights in summarizing documents written in languages other than English (Section 4.5), and (iv) analyze the impact of the main MWI-Sum algorithm parameters and features on the summarization performance (Section 4.6). Finally, we also analyzed the impact of the collection size on the quality of the generated summaries (Section 4.7). All the experiments were performed on a 3.0GHz 64 bit Intel Xeon PC with 4GB main memory running Ubuntu 10.04 LTS (kernel 2.6.32-31).

4.1. Textual Document Collections

We evaluated MWI-Sum performance on (i) the MultiLing pilot TAC'11 datasets [Giannakopoulos et al. 2011], which are the reference collections of the TAC'11 multilingual summarization task [Text Analysis Conference 2011], (ii) the task 2 datasets of DUC'04 [Document Understanding Conference 2004], which are English-written benchmark collections commonly used to evaluate generic multidocument summarization algorithms, and (iii) 11 real-life multilingual news article collections retrieved in April 2012. A short description of the used collections follows.

4.1.1. TAC 2011 Multilingual Benchmark Collections. The MultiLing pilot [Giannakopoulos et al. 2011] is the multilingual summarization task of TAC'11 [Text Analysis Conference 2011]. TAC'11 documents are clustered into 10 groups. Each group contains 10 documents. Document groups have been translated in seven different languages (Arabic, Czech, English, French, Greek, Hebrew, Hindi) by native speakers. For each document group and language, at least one golden summary (i.e., the optimal document collection summary) is given. Among the available languages, we tested all the European (i.e., English, French, Czech, and Greek), Arabic, and Hindi languages.

4.1.2. DUC 2004 English-written Benchmark Collections. The dataset of Task 2 of the DUC'04 competition [Document Understanding Conference 2004] is a commonly used benchmark in the context of generic English-written multidocument summarization. DUC'04 documents are clustered into 50 document groups, which include 10 English-written documents each. At least one golden summary is given for each of the DUC'04 document groups.

4.1.3. News Collections. In April 2012, we retrieved from the Web 11 different news document collections, six written in English and five in Italian. Each collection is associated with a different topic and is composed of 10 documents (news). Documents within each collection range over the same topic. Each collection was collected by providing a query, focused on a given topic, to the Google News search engine and then by selecting the 10 top-ranked news articles. For each language, its ad hoc Google News version was used. The queries addressed the following topics:

—**Debt crisis**: the financial debt crisis in the United States and European countries
—**London 2012**: the Olympic Games held in August 2012 in London (UK)
—**French elections 2012**: the French presidential elections held on April 22, 2012
—**Tsunami in Japan 2011**: the tsunami caused by a violent earthquake that hit Japan on April 7, 2011
—**Organic farming**: the technique of agriculture based on biological pest control
—**Hurricane Irene 2011**: Hurricane Irene beats down on the U.S. East Coast in 2011

The topics were selected as representatives of different case studies: (i) very focused news of topical interest (e.g., the French elections 2012, Hurricane Irene 2011), (ii) averagely focused past events (e.g., tsunami in Japan 2011, London 2012), (iii) broad-spectrum and multifaceted news (e.g, debt crisis), and (iv) broad-spectrum and long-term matter of contention (e.g, organic farming).

The retrieved news collections are available at http://dbdmg.polito.it/wordpress/research/document-summarization/.

4.2. Performance Comparison on Multilingual Datasets

This section presents the evaluation of MWI-Sum multilingual performance on the MultiLing pilot TAC'11 benchmark dataset (see Section 4.1.1). We compared the performance of our approach on the TAC'11 collections with (i) all the summarization methods submitted to the TAC'11 conference; (ii) a recently proposed summarizer relying on word association discovery, i.e., Association Mixture Text Summarization (AMTS) [Gross et al. 2014]; (iii) three widely used open-source text summarizers, i.e., the ILP-based ICSI multidocument summarization system (ICSIsumm) [Gillick et al. 2009, 2008], the Open Text Summarizer (OTS) [Rotem 2011], and TexLexAn [TexLexAn 2011]; and (iv) ItemSum (Itemset-based Summarizer), a preliminary version of the MWI-Sum summarizer presented in Baralis et al. [2012]. Since the source code of the AMTS summarizer was not publicly available on the Web, we reimplemented the summarizer to the best of our understanding based on the indications given in the reference article [Gross et al. 2014].

To evaluate the effectiveness of MWI-Sum and its competitors on multilingual collections, we summarized separately the collections written in different languages by means of the same summarizer. For the TAC'11 competitors, we considered the summaries provided by the TAC'11 system [Text Analysis Conference 2011]. To compare MWI-Sum with the other approaches, we used the ROUGE toolkit [Lin and Hovy 2003], which has been adopted as official TAC'11 and DUC'04 tool for performance evaluation,[1] but also JRouge [Krapivin et al. 2014] and AutoSummENG[2] [Giannakopoulos and Karkaletsis 2011], which are considered more adequate for non-English summary evaluation, in particular when Unicode is used to code textual contents. All three softwares we used measure the quality of a summary by counting, by means of different metrics, the unit overlaps between the candidate summary and a set of reference summaries (i.e., the golden summaries) provided by the TAC'11 and DUC'04 organizers. Intuitively, the summarizer that achieves the highest scores can be considered the most effective. To perform a fair comparison, before using the evaluation tools, the generated summaries have been normalized by truncating each of them at 665 bytes (rounding the number down in case of straddled words), following the same approach used in the DUC'04 competition [Document Understanding Conference 2004]. Several automatic evaluation scores are implemented in ROUGE, JRouge, and AutoSummENG. For the sake of brevity, we only report the results for ROUGE-2 and ROUGE-SU4 for Rouge, which are considered the most representative scores [Lin and Hovy 2003], ROUGE-2 for JRouge,[3] and Average Similarity for AutoSummENG. To successfully evaluate non-English documents, we temporarily disabled the stemming and stopword preprocessing steps embedded in the ROUGE evaluation toolkit, and we provided as input to the evaluator the previously preprocessed (i.e., truncated) summaries.

For the TAC'11 competitors each of the considered summaries was generated by the best algorithm configuration for the corresponding language and documents. Hence, for the other tested summarizers, we used the algorithm configuration suggested by the respective authors for each language and collection tested. Since ItemSum has never been tested in a multilingual context, we used the best configuration reported in Baralis et al. [2012] (support threshold $minsup = 3\%$, model size $ms = 12$). To perform a fair comparison, for AMTS, we did not integrate any external dictionary into

[1] We used the command: ROUGE-1.5.5.pl -e data -x -m -2 4 -u -c 95 -r 1000 -n 4 -f A -p 0.5 -t 0 -d -a.

[2] We used the same setting recommended in the TAC'11 contest, i.e., $L_{min} = 3$, $L_{max} = 3$, and $D_{win} = 3$.

[3] ROUGE-SU4 is not available in JRouge.

Table V. TAC'11 Multilingual Collections: Borda Count Ranking Achieved by MWI-Sum and the Other Approaches over All the Tested Languages

	Borda Count Ranking			
	JRouge	AutoSummENG	ROUGE	
Summarizer	ROUGE-2 F1	Avg. Similarity	ROUGE-2 F1	ROUGE-SU4 F1
MWI-Sum	1st	1st	1st	1st
JRC	2nd	2nd	5th	4th
CLASSY	3rd	3rd	3rd	3rd
ItemSum	4th	5th	4th	5th
AMTS	5th	8th	6th	6th
ICSIsumm	6th	6th	2nd	2nd
CIST	7th	4th	8th	8th
LIF	8th	7th	7th	7th
UBSummarizer	9th	9th	9th	9th

Table VI. TAC'11 Multilingual Collections: Arabic-Written Collections
Evaluation by means of JRouge and AutoSummENG. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 1%, $K = 4$, stopword elimination) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
MWI-Sum	0.0719	0.2793	0.1144	0.0871
AMTS	0.0677	0.2650	0.1078	0.0826
JRC	0.0670	0.2678	0.1072	0.0842*
UoEssex	0.0653	0.2520	0.1036	0.0817
ItemSum	0.0598	0.2250	0.0944	0.0780
LIF	0.0528	0.2070	0.0841	0.0707*
CLASSY	0.0513	0.2029	0.0819	0.0686*
TALN_UPF	0.0450*	0.1716*	0.0713*	0.0675*
CIST	0.0336*	0.1307*	0.0534*	0.0587*
ICSIsumm	0.0297*	0.2128	0.0512*	0.0360*
UBSummarizer	0.0176*	0.0675*	0.0279*	0.0440*

the document summarization process. Similar to what previously done by the other multilingual summarizers (e.g., Conroy et al. [2011]), for MWI-Sum, we tailored the standard configuration (i.e., the values of minimum support threshold $\mathcal{W}$minsup and number of top relevant sentences K) to the language under analysis. Specifically, for each language, we followed an approach similar to the one used in Takamura and Okumura [2009] , and we considered as standard configuration the optimal one tuned on ROUGE-2. For each tested language, the MWI-Sum standard configuration used in the performed experiments is reported in the captions of Tables XII–XVII. While stopword elimination is applied to documents written in all languages, stemming is selectively enabled/disabled according to the language in which the analyzed documents are written. Specifically, on documents written in non-European languages (i.e., Arabic and Hindi) and in languages based on non-Latin alphabets (i.e., Greek), the stemmer was disabled because it appears to worsen summarizer performance. Conversely, on languages based on the Latin alphabet (i.e., Czech, English, and French) the stemmer was enabled because it yields significant performance improvements. A thorough discussion of the impact of the MWI-Sum configuration parameters on the summarization performance is given in Section 4.6.

Tables VI–XI report the results achieved by using the JRouge and AutoSummENG toolkits for evaluation. More specifically, they report the comparisons, in terms of ROUGE-2 and ROUGE-S4 Precision (P), Recall (R), and F1-measure (F1) (computed

Table VII. TAC'11 Multilingual Collections: Czech-Written Collections
Evaluation by means of JRouge and AutoSummENG. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 0.75%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
CIST	**0.0710**	**0.1928**	**0.1037**	**0.1033**
CLASSY	0.0692	0.1849	0.1006	0.0952
MWI-Sum	0.0656	0.1772	0.0957	0.0966
ItemSum	0.0627	0.1751	0.0923	0.0912
JRC	0.0627	0.1752	0.0923	0.0964
AMTS	0.0557	0.2441	0.0898	0.0684
LIF	0.0529	0.1514	0.0783	0.0817
OTS	0.0527	0.1398	0.0765	0.0806
ICSIsumm	0.0504	0.1481	0.0750	0.0864
UBSummarizer	0.0297*	0.0806*	0.0433*	0.0631

Table VIII. TAC'11 Multilingual Collections: English-Written Collections
Evaluation by means of JRouge and AutoSummENG. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 1%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
MWI-Sum	**0.1086**	**0.2382**	**0.1490**	0.1328
ICSIsumm	0.0966	0.2191	0.1340	0.1297
JRC	0.0959	0.2224	0.1339	**0.1343**
ItemSum	0.0927	0.2137	0.1293	0.1211
CLASSY	0.0895	0.2038	0.1244	0.1166
LIF	0.0801*	0.1809*	0.1109*	0.1073*
OTS	0.0778	0.1798	0.1085	0.1066
TALN_UPF	0.0770*	0.1719*	0.1063*	0.1086
CIST	0.0740	0.1663	0.1024	0.1178
TexLexAn	0.0713*	0.1666*	0.0998*	0.1037*
UoEssex	0.0703	0.1625	0.0981	0.1052
SIEL_IIITH	0.0682*	0.1527*	0.0941*	0.0973*
AMTS	0.0578*	0.1973	0.0881*	0.0782*
UBSummarizer	0.0445*	0.0981*	0.0612*	0.0910

by JRouge) between our approach and the other considered approaches on the TAC'11 Arabic-, English-, French-, Greek-, Czech-, and Hindi-written collections and the comparison in terms of average similarity (computed by AutoSummENG). The summarizers are ranked in order of decreasing ROUGE-2 F1-measure. Any statistically relevant difference in the comparisons between MWI-Sum and its competitors, evaluated by the paired t-test [Dietterich 1998] at 95% significance level, is starred in Tables VI–XI.

The overall ranking in terms of multilingual summarizer performance is reported in Table V. We reported the achieved overall rankings for each of the considered measures (the summarizers are ranked in order of decreasing ROUGE-2 F1 ranking computed by JRouge). To produce a unique ranking, for each of the considered measures, we first computed the average ranking achieved by each summarizer that succeeds in summarizing documents written in all the considered languages. Then, to properly combine the individual rankings that were achieved on different languages into a unified ranking list, for each evaluation measure, we adopted an established aggregation function based on the Borda Count group consensus function [van Erp and Schomaker 2000]. This approach first assigns decreasing integer scores to the elements of each individual rank and then it combines the voting scores to generate a unique ranking. To

Table IX. TAC'11 Multilingual Collections: French-Written Collections
Evaluation by means of JRouge and AutoSummENG. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 0.9%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
MWI-Sum	**0.1028**	**0.2332**	**0.1426**	0.1302
ICSIsumm	0.0978	0.2258	0.1363	0.1321
CLASSY	0.0973	0.2184	0.1346	0.1142
JRC	0.0954	0.2239	0.1337	**0.1327**
AMTS	0.0938	0.2310	0.1331	0.1145
LIF	0.0886	0.2061	0.1238	0.1123
CIST	0.0871	0.1977	0.1208	0.1210
ItemSum	0.0847	0.1938	0.1179	0.1104
OTS	0.0826	0.1848	0.1140	0.1028*
UBSummarizer	0.0696*	0.1574*	0.0965*	0.1142
TexLexAn	0.0669*	0.1368*	0.0898*	0.0918*
SIEL_IIITH	0.0615*	0.1369*	0.0848*	0.0926*
TALN_UPF	0.0603*	0.1377*	0.0838*	0.0984*

Table X. TAC'11 Multilingual Collections: Greek-Written Collections
Evaluation by means of JRouge and AutoSummENG. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 1.2%, $K = 3$, stopword elimination) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
CLASSY	**0.0479**	**0.2097**	**0.0778**	**0.0663**
MWI-Sum	0.0418	0.1900	0.0684	0.0649
JRC	0.0383	0.1737	0.0626	0.0643
AMTS	0.0376	0.1622	0.0610	0.0604
ICSIsumm	0.0349	0.1443	0.0562	0.0559
ItemSum	0.0326	0.1483	0.0534	0.0512
LIF	0.0301*	0.1311*	0.0489*	0.0521
OTS	0.0297*	0.1347*	0.0486*	0.0490*
CIST	0.0266*	0.1153*	0.0432*	0.0516*
UBSummarizer	0.0143*	0.0629*	0.0233*	0.0399*

Table XI. TAC'11 Multilingual Collections: Hindi-Written Collections
Evaluation by means of JRouge and AutoSummENG. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 3.5%, $K = 3$, stopword elimination) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
MWI-Sum	**0.0984**	**0.4709**	**0.1622**	**0.0619**
JRC	0.0983	0.4650	0.1617	0.0521
CLASSY	0.0932	0.4551	0.1541	0.0507
CIST	0.0913	0.4434	0.1509	0.0480
SIEL_IIITH	0.0876	0.4281	0.1451	0.0441*
AMTS	0.0864	0.4936	0.1444	0.0425*
ItemSum	0.0852	0.4130	0.1407	0.0453
TALN_UPF	0.0747*	0.3773*	0.1244*	0.0375*
UBSummarizer	0.0701*	0.3524*	0.1165*	0.0319*
LIF	0.0627*	0.3145*	0.1041*	0.0311*
ICSIsumm	0.0279*	0.1651*	0.0470*	0.0139*

Table XII. TAC'11 Multilingual Collections: Arabic-Written Collections
Evaluation by means of the the ROUGE toolkit. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 1%, $K = 4$, stopword elimination) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
UoEssex	0.0834	0.1289	**0.0982**	**0.0993**	0.1983	**0.1263**
AMTS	0.0812	0.1316	0.0969	0.0968	0.2020	0.1247
ItemSum	**0.0851**	0.1227	0.0952	0.0883	0.1526	0.1022
LIF	0.0717	0.1135	0.0775	0.0769	0.1535	0.0863
MWI-Sum	0.0513	**0.2210**	0.0756	0.0569	**0.2908**	0.0839
CLASSY	0.0605	0.1221	0.0719	0.0543	0.2528	0.0711
ICSIsumm	0.0487	0.1035	0.0626	0.0523	0.1216	0.0665
CIST	0.0502	0.0708	0.0578	0.0378	0.0825	0.0482
JRC	0.0454	0.1338	0.0519	0.0633	0.1965	0.0714
TALN_UPF	0.0337	0.0702	0.0445	0.0418	0.1362	0.0602
UBSummarizer	0.0092*	0.0671*	0.0161*	0.0107*	0.1375*	0.0195*

Table XIII. TAC'11 Multilingual Collections: Czech-Written Collections
Evaluation by means of the the ROUGE toolkit. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 0.75%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
MWI-Sum	**0.0936**	0.2576	**0.1372**	0.0975	0.2710	0.1433
CIST	0.0924	0.2567	0.1359	**0.1001**	0.2807	**0.1475**
CLASSY	0.0887	0.2477	0.1306	0.0960	0.2697	0.1415
JRC	0.0867	0.2465	0.1282	0.0947	0.2716	0.1403
ItemSum	0.0840	0.2460	0.1252	0.0905	0.2653	0.1349
ICSIsumm	0.0763	0.2206	0.1132	0.0872	0.2534	0.1296
AMTS	0.0697	**0.3087**	0.1125	0.0718	**0.3156**	0.1158
LIF	0.0748	0.2173	0.1113	0.0816	0.2390	0.1216
OTS	0.0719	0.1997	0.1057	0.0826	0.2313	0.1217
UBSummarizer	0.0444*	0.1276*	0.0658*	0.0638*	0.1840*	0.0947*

effectively deal with ranking lists of different lengths, in our Borda Count implementation, we assign to the first element of each rank the same value equal to the length of the longest input rank. Based on the Borda Count method, MWI-Sum ranks first in terms of ROUGE-2 and average similarity-Measure averaged over all the tested languages and collections (Table V).

On individual languages, MWI-Sum ranks first in terms of ROUGE-2 on Arabic, English, French, and Hindi; second on Greek; and third on Czech (Tables VI–XI). Similar results are achieved in terms of average similarity computed by means of AutoSummENG. More specifically, MWI-Sum ranks first on Arabic and Hindi; second on English, Czech, and Greek; and third on French.

For completeness, Tables XII–XVII report the results achieved by using the ROUGE toolkit [Lin and Hovy 2003], which was one of the two official TAC'11 evaluation tools.[4] On the English document collections, which do not contain particular unicode characters, the results achieved by the ROUGE toolkit and JRouge, in terms of the common ROUGE-2 measure, are exactly the same. Differently, on non-English documents, the results are different. However, also by considering the results achieved by applying the

[4] The other evaluation tool used at TAC'11 was AutoSummENG.

Table XIV. TAC'11 Multilingual Collections: English-Written Collections

Evaluation by means of the the ROUGE Toolkit. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 1%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
MWI-Sum	**0.1086**	**0.2382**	**0.1490**	**0.1204**	**0.2669**	**0.1658**
ICSIsumm	0.0966	0.2191	0.1340	0.1105	0.2526	0.1537
JRC	0.0959	0.2224	0.1339	0.1053	0.2467	0.1475
ItemSum	0.0927	0.2137	0.1293	0.1056	0.2446	0.1474
CLASSY	0.0895	0.2038	0.1244	0.1076	0.2468	0.1498
LIF	0.0801*	0.1809*	0.1109*	0.0978*	0.2239*	0.1361*
OTS	0.0778	0.1798	0.1085	0.0936	0.2172	0.1307
TALN_UPF	0.0770*	0.1719*	0.1063*	0.0984	0.2210	0.1361
CIST	0.0740	0.1663	0.1024	0.0915	0.2080	0.1271
TexLexAn	0.0713*	0.1666*	0.0998*	0.0891*	0.2101	0.1251*
UoEssex	0.0703	0.1625	0.0981	0.0848	0.1981	0.1187
AMTS	0.0700	0.1629	0.0979	0.0849	0.1997	0.1191
SIEL_IIITH	0.0682*	0.1527*	0.0941*	0.0859*	0.1941*	0.1189*
UBSummarizer	0.0445*	0.0981*	0.0612*	0.0725*	0.1604*	0.0998*

Table XV. TAC'11 Multilingual Collections: French-Written Collections

Evaluation by means of the the ROUGE toolkit. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 0.9%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
MWI-Sum	**0.1071**	**0.2479**	**0.1494**	**0.1183**	**0.2759**	**0.1654**
JRC	0.1029	0.2399	0.1439	0.1114	0.2622	0.1562
ICSIsumm	0.1021	0.2400	0.1432	0.1146	0.2718	0.1611
CLASSY	0.1024	0.2337	0.1424	0.1112	0.2563	0.1550
AMTS	0.0961	0.2446	0.1375	0.1052	0.2692	0.1508
CIST	0.0947	0.2175	0.1318	0.1100	0.2544	0.1534
LIF	0.0905	0.2169	0.1276	0.1057	0.2556	0.1495
ItemSum	0.0891	0.2082	0.1248	0.1018	0.2396	0.1429
TexLexAn	0.0901	0.2026	0.1247	0.1008*	0.2279*	0.1397*
OTS	0.0818*	0.1914	0.1145*	0.0931*	0.2181*	0.1304*
UBSummarizer	0.0729*	0.1693*	0.1019*	0.0926*	0.2165*	0.1297*
SIEL_IIITH	0.0621*	0.1465*	0.0871*	0.0838*	0.1978*	0.1176*
TALN_UPF	0.0619*	0.1451*	0.0867*	0.0859*	0.2040*	0.1209*

ROUGE toolkit, MWI-Sum is, on average, better than the other approaches (see the results in terms of Borda Count Ranking reported in the last two columns of Table V). On individual languages, MWI-Sum ranks first in terms of ROUGE-2 on English, French, and Czech; second on Hindi; fourth on Greek; and fifth on Arabic (see Tables XII–XVII). Similar results are achieved in terms of ROUGE-SU4. Specifically, MWI-Sum ranks first in terms of ROUGE-SU4 on English, French, and Hindi; second on Czech; third on Greek; and fourth on Arabic.

We also tested the performance of MWI-Sum on TAC'11 collections by disabling the initial stemming step also on English, French, and Czech. MWI-Sum performance slightly degrades by disabling stemming on these languages, but the performance decrease with respect to MWI-Sum with enabled stemming is, in most cases, not statistically significant.

Table XVI. TAC'11 Multilingual Collections: Greek-Written Collections
Evaluation by means of the the ROUGE toolkit. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 1.2%, $K = 3$, stopword elimination) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
ICSIsumm	**0.0940***	0.1673	**0.1167**	**0.0904***	0.2004	**0.1161**
CLASSY	0.0749	0.1961	0.1032	0.0662	0.2141	0.0937
AMTS	0.0659	0.1316	0.0850	0.0787	0.1879	0.1047
MWI-Sum	0.0473	**0.2233**	0.0751	0.0462	**0.3361**	0.0744
ItemSum	0.0531	0.1139	0.0701	0.0441	0.1327	0.0623
LIF	0.0466	0.0717*	0.0548	0.0543	0.1040*	0.0679
OTS	0.0353	0.1443	0.0545	0.0298	0.1658	0.0467
JRC	0.0240	0.0834	0.0363	0.0241	0.0924*	0.0356*
CIST	0.0229	0.0890*	0.0348	0.0243	0.1322	0.0378*
UBSummarizer	0.0065*	0.0331*	0.0109*	0.0119*	0.1369	0.0209*

Table XVII. TAC'11 Multilingual Collections: Hindi-Written Collections
Evaluation by means of the the ROUGE toolkit. Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 3.5%, $K = 3$, stopword elimination) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
ICSIsumm	**0.0500**	**0.0501**	**0.0486**	0.0673	0.0786	0.0681
MWI-Sum	0.0417	0.0477	0.0404	**0.0760**	**0.1000**	**0.0688**
ItemSum	0.0498	0.0332	0.0398	0.0356	0.0192	0.0249
JRC	0.0249	0.0166	0.0199	0.0412	0.0651	0.0326
SIEL_IIITH	0.0084	0.0112	0.0096	0.0206	0.0335	0.0255
AMTS	0	0	0	0.0142	0.0052	0.0077
CIST	0	0	0	0	0	0
CLASSY	0	0	0	0	0	0
LIF	0	0	0	0	0	0
UBSummarizer	0	0	0	0	0	0
TALN_UPF	0	0	0	0	0	0

Finally, we also analyzed the average number of words per sentence. Table XVIII reports the statistics obtained by all the considered summarizers. The average number of words per sentence of MWI-Sum is similar to those of the majority of the other approaches except for CIST, which selects sentences that are on average longer than those of all the others.

Note that the average sentence length and the total size of the generated summaries do not affect the evaluation results. To fairly compare summaries with the ROUGE toolkit, we truncated summaries at 665 bytes following the same approach used in the DUC'04 competition [Document Understanding Conference 2004]. This preprocessing step reduces the bias in the comparison between summaries of different length.

4.3. Performance Comparison on English-Written Datasets

We compared the performance of our approach on the English-written DUC'04 Benchmark collection with that of (i) all the 35 summarization methods submitted to the DUC'04 conference, (ii) the eight summaries generated by humans and provided by the DUC'04 system, (iii) the Association Mixture Text Summarization (AMTS) tool [Gross et al. 2014], (iv) the ICSI multidocument summarization system (ICSIsumm) [Gillick et al. 2008, 2009], (v) the Open Text Summarizer (OTS) [Rotem

Table XVIII. TAC'11 Multilingual Collections: Statistics on the Average Number of Words Per Sentence

	Language					
Summarizer	Arabic	English	French	Greek	Czech	Hindi
MWI-Sum	23.0	25.7	30.7	24.4	22.5	34.0
JRC	25.2	25.7	26.2	26.1	23.0	28.9
CLASSY	20.6	20.9	20.8	19.5	18.2	31.3
ItemSum	35.7	34.7	44.3	43.0	28.8	54.9
AMTS	14.7	14.5	17.9	16.3	9.7	20.6
ICSIsumm	27.5	21.9	25.9	28.9	17.9	79.3
CIST	54.9	45.0	36.7	55.3	42.4	48.7
LIF	15.3	16.2	21.1	13.3	11.3	14.2
UBSummarizer	22.9	23.0	22.7	23.1	22.8	22.5
TALN_UPF	27.9	32.9	37.0	-	-	29.8
OTS	-	15.6	20.0	16.9	16.0	-
TexLexAn	-	7.9	3.4	-	-	-
SIEL_IIITH	-	14.3	22.1	-	-	14.1
UoEssex	17.5	16.1	-	-	-	-

Table XIX. DUC'04 Collections

Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}$minsup = 0.6%, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
DUC'04 CLASSY-Serif peer67	0.0906	**0.0941**	**0.0922**	0.1313	**0.1362**	**0.1335**
DUC'04 CLASSY-pre peer65	**0.0922**	0.0909	0.0915	**0.1335**	0.1313	0.1323
MWI-Sum	0.0904	0.0916	0.0909	0.1312	0.1328	0.1319
DUC'04 CLASSY-baseline peer66	0.0888	0.0936	0.0909	0.1284	0.1352	0.1313
ICSIsumm	0.0877	0.0861	0.0869	0.1310	0.1285	0.1297
ItemSum	0.0852	0.0870	0.0859	0.1254	0.1276	0.1262
DUC'04 peer102	0.0848	0.0859	0.0853	0.1273	0.1286	0.1278
DUC'04 peer104	0.0857	0.0842*	0.0849	0.1294	0.1270	0.1281
DUC'04 peer35	0.0837	0.0842	0.0839	0.1288	0.1297	0.1292
DUC'04 peer124	0.0833*	0.0819*	0.0826*	0.1278	0.1253*	0.1265
DUC'04 peer19	0.0803*	0.0804*	0.0803*	0.1247*	0.1247*	0.1246*
DUC'04 peer81	0.0808*	0.0790*	0.0799*	0.1253	0.1224*	0.1238*
DUC'04 peer34	0.0763*	0.0764*	0.0763*	0.1236*	0.1240*	0.1238*
OTS	0.0744*	0.0740*	0.0742*	0.1151*	0.1144*	0.1147*
TexLexAn	0.0658*	0.0655*	0.0656*	0.1096*	0.1088*	0.1092*
AMTS	0.0635*	0.0651*	0.0642*	0.1014*	0.1040*	0.1025*

2011], (vi) TexLexAn [TexLexAn 2011], and (vii) ItemSum [Baralis et al. 2012]. For the DUC'04 competitors, the results provided by the DUC'04 system [Document Understanding Conference 2004] are considered.

Since the documents are written in English, and hence special unicode characters do not occur in this context, for the sake of brevity, we used the ROUGE toolkit to evaluate the generated summaries. We decided to use the ROUGE toolkit instead of JRouge because the ROUGE toolkit allows computing also the ROUGE-SU4 measure, that is usually considered one of the most representative scores [Lin and Hovy 2003].

Table XIX compares the performance of the MWI-Sum summarizer on the DUC'04 dataset with that of ICSIsumm, ItemSum, OTS, TexLexAn, and the 10 most effective DUC'04 peers in terms of ROUGE-2 Precision (P), Recall (R), F1 (F1), and average similarity. MWI-Sum ranked third in terms of both ROUGE-2 F1 and ROUGE-SU4 F1 out of the 39 considered methods. It performs better than ICSIsumm, ItemSum, OTS, and TexLexAn on both ROUGE-2 and ROUGE-SU4. Only the CLASSY summarizer [Conroy et al. 2004], whose three variations (CLASSY-pre, CLASSY-baseline, and CLASSY-Serif) were submitted to DUC'04 as peers 65, 66, and 67, in some cases performs better than MWI-Sum. However, unlike MWI-Sum, CLASSY [Conroy et al. 2004] heavily relies on language-dependent linguistic analysis. Thus, as discussed in the previous section, it shows a more limited portability to languages other than English.

In Table XIX, statistically relevant differences in the comparisons between MWI-Sum and the other approaches, evaluated by the paired t-test [Dietterich 1998] at 95% significance level, are starred. The performance improvements that were achieved by MWI-Sum with respect to OTS and TexLexAn are statistically significant for every evaluator and measure. MWI-Sum also performs significantly better than three of the top-10 DUC'04 methods. Furthermore, MWI-Sum is never significantly outperformed by any of the other summarizers in terms of ROUGE-2 and ROUGE-SU4 F1-measure. Hence, MWI-Sum performs as good as the best DUC'04 summarizers on English-written documents without performing any advanced linguistic analysis.

We also compared the summaries generated by MWI-Sum and the other considered summarizers with the humanly generated summaries provided by the DUC'04 organizers. MWI-Sum and CLASSY-Serif (i.e., the most effective version of the CLASSY summarizer [Conroy et al. 2004] in terms of ROUGE-SU4 F1-measure) are the only two summarizers that performed better than humans on any DUC'04 collection.

4.4. Results on Multilingual News Documents

We evaluated the performance of the MWI-Sum system on real-life news document collections (see Section 4.1.3). We compared the performance of our approach with the ones of different publicly available summarizers. Specifically, we considered (i) three widely used open-source text summarizers, i.e., the ILP-based ICSI multi-document summarization system (ICSIsumm) [Gillick et al. 2009, 2008], the Open Text Summarizer (OTS) [Rotem 2011], and TexLexAn [TexLexAn 2011]; (ii) the Association Mixture Text Summarization (AMTS) tool [Gross et al. 2014]; and (iii) ItemSum (Itemset-based Summarizer), a preliminary version of the MWI-Sum summarizer presented in Baralis et al. [2012].

Since golden summaries are not available for the crawled news, to evaluate the summarization performance we adopted a leave-one-out cross-validation, as previously done in Chuang and Yang [2000]. More specifically, for each category and language, we summarized 9 out of 10 news documents, and we compared the achieved summary with the remaining (not yet considered) one, which was selected as golden summary. Next, we tested all the other possible combinations by varying the golden summary, and for each summarizer, we computed the average performance results, in terms of precision (P), Recall (R), and F1-measure (F1), for both the ROUGE-2 and ROUGE-SU4 evaluation scores. Since we specifically cope with documents ranging over the same topic, the assumption that a document is a representative summary of all the other documents in the collection is acceptable.

This section is organized as follows. Section 4.4.1 discusses the summaries generated by MWI-Sum and the other summarizers on an English-written news article collection chosen as representative, while Section 4.4.2 presents the results of the performance comparison between MWI-Sum and the other summarizers on all the considered news collections.

4.4.1. Summary Comparison. Table XX reports the top-3 sentences of the summaries generated by MWI-Sum and the other summarizers on the English-written Hurricane Irene news article collection, chosen as representative of all the considered collections.[5] The summary generated by MWI-Sum provides an exhaustive and non-redundant overview of the analyzed topic. {*Irene, Hurricane, New, York*}, {*disaster, government*}, and {*tropical, storm*} are examples of weighted itemsets included in the itemset-based model generated by MWI-Sum, which are covered by the top-3 summary sentences. The summary generated by ItemSum also shows a relatively good quality. However, it includes a sentence, i.e., *"If this is what it means to live in the nanny state, I'm very content" Krasnow said*, that provides marginal information about the news of interest. Even the summaries generated by ICSIsumm and AMTS cover some important facets of the topic. However, they also include sentences that seem to be too specific and of interest for a limited period of time (e.g., *In Vermont, officials planned to airlift food and water to towns cut off by the floodwaters.*). OTS and TexLexAn extract very focused sentences, which provide rather peculiar information.

In Column 1 of Table XX, we reported, for each method, its name and the size of its summary (in terms of number of sentences). Both MWI-Sum and OTS generated summaries composed of 14 sentences, ICSIsumm generated a summary consisting of 9 sentences, whereas ItemSum and AMTS included three sentences in their summary. Only TexLexAn generated a very long summary with 54 sentences. However, note that while MWI-Sum, similarly to ItemSum, OTS and TexLexAn, automatically selects the size of the generated summaries, ICSIsumm generates summaries with a size equal to a user-specified value (the summary size is a mandatory parameter of ICSIsumm). In the performed experiment, we set the expected ICSIsumm summary size to 250 words, which obviously had an impact on the number of sentences of its summary.

4.4.2. Performance Comparison on News Documents. We evaluated the performance of the MWI-Sum summarizer on all the news document collections (see Section 4.1.3). Tables XXI–XXIV report the average results that were obtained by MWI-Sum and by the other considered summarizers (i.e., ICSIsumm, AMTS, ItemSum, OTS, and TexLexAn) on the English- and Italian-written news document collections. For all the considered datasets and measures, the statistical significance of the performance difference between MWI-Sum and the other approaches was evaluated by the paired t-test [Dietterich 1998] at 95% significance level. Statistically relevant differences are starred in Tables XXI–XXIV. For each considered dataset and measure, the results that were achieved by the most effective summarizers are written in boldface.

MWI-Sum performs statistically better than ItemSum, OTS, AMTS, and TexLexAn on the English-written collections (see Tables XXI–XXII). It also performs statistically better than ICSIsumm in terms of ROUGE-SU4 and Average Similarity on the same collection. On the Italian collection, MWI-Sum ranked first independently of the considered evaluation metric. However, the differences are statistically relevant only with respect to ItemSum (see Tables XXIII–XXIV) in terms of ROUGE-2 and ROUGE-SU4 and with respect to OTS in terms of Average Similarity (Table XXIII).

4.5. Effect of Item Weights in a Multilingual Context

The MWI-Sum summarizer addresses the multilingual document summarization problem by exploiting weighted itemsets. Previous approaches (i.e., ItemSum [Baralis et al. 2012]) already exploited traditional (unweighted) itemsets to summarize English-written documents. To analyze the effect of item weights in the summarization of

[5] Most of the considered summaries consist of more than three sentences. However, we focused our analyses on the first three sentences because the first selected ones are the most representative of each summary.

Table XX. Summary Examples: English-Written Hurricane Irene News Collection

Method	Summary (top-3 sentences)
MWI-Sum (total summary length: 14 sentences)	New York: Hurricane Irene will most likely prove to be one of the 10 costliest catastrophes in the nation's history, and analysts said that much of the damage might not be covered by insurance because it was caused not by winds but by flooding, which is excluded from many standard policies. As emergency airlift operations brought ready-to-eat meals and water to Vermont residents left isolated and desperate, states along the Eastern Seaboard continued to be battered Tuesday by the after effects of Irene, the destructive hurricane turned tropical storm. Obama and several politicians with possible White House designs of their own faced that challenge last week: to avoid repeating President George W. Bush's mistake in 2005 of appearing disengaged from the government response to an impending natural disaster.
ItemSum (total summary length: 3 sentences)	New York was pounded by heavy winds and torrential rain on Sunday morning as Hurricane Irene bore down on the city, threatening to cause flash flooding and widespread damage in the US's most populous city. "If this is what it means to live in the nanny state, I'm very content," Krasnow said. "People and families coping with these natural disasters will certainly get what they need from the federal government, but the goal should be to find ways to pay for what is needed when possible," Cantor's office said in a memo.
ICSIsumm (total summary length: 9 sentences)	With estimated damage from Hurricane Irene topping $7 billion, the White House and some in Congress are at odds over where to find money to replenish the disaster relief fund of the Federal Emergency Management Agency, which has dipped below the $1 billion level considered advisable. In Vermont, officials planned to airlift food and water to towns cut off by the floodwaters. "This is not over," Obama said Sunday afternoon as the storm battered New England.
AMTS (total summary length: 9 sentences)	With estimated damage from Hurricane Irene topping $7 billion, the White House and some in Congress are at odds over where to find money to replenish the disaster relief fund of the Federal Emergency Management Agency, which has dipped below the $1 billion level considered advisable. On Sunday, FEMA stirred some controversy when it announced that to meet the current crisis it was temporarily suspending payments for rebuilding in areas hit by earlier disasters, such as this spring's tornadoes in Missouri and other states. Then on Monday, House Majority Leader Eric Cantor told a Fox News audience that any new federal disaster monies would require offsetting cuts in other spending, igniting a round of budgetary who-goes-first.
OTS (total summary length: 14 sentences)	As emergency airlift operations brought ready-to-eat meals and water to Vermont residents left isolated and desperate, states along the Eastern Seaboard continued to be battered Tuesday by the after effects of Irene, the destructive hurricane turned tropical storm. Dangerously damaged infrastructure, 2.5 million people without power and thousands of water-logged homes and businesses continued to overshadow the lives of residents and officials from North Carolina through New England, where the storm has been blamed for at least 44 deaths in 13 states. But new dangers developed in New Jersey and Connecticut, where once benign rivers rose menacingly high.
TexLexAn (total summary length: 54 sentences)	As emergency airlift operations brought ready-to-eat meals and water to Vermont residents left isolated and desperate, states along the Eastern Seaboard continued to be battered Tuesday by the after effects of Irene, the destructive hurricane turned tropical storm. Search-and-rescue teams in Paterson have pulled nearly 600 people from flooded homes in the town after the Passaic River rose more than 13 feet above flood stage, the highest level since 1903. It's one of several towns in states such as New Jersey, Connecticut, New York, Vermont and Massachusetts dealing with the damage of torrential rain and flooding spawned by Hurricane Irene (Cleveland..Hurricane Katrina was a Category 5 hurricane and Hurricane Irene was a Category 1 hurricane.

Table XXI. English News Collections: Evaluation by Means of JRouge and AutoSummENG
Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}minsup = 0.8\%$, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
MWI-Sum	**0.0185**	**0.0998**	**0.0296**	**0.0434**
ICSIsumm	0.0179	0.0985	0.0289	0.0396*
ItemSum	0.0151*	0.0919*	0.0238*	0.0350*
AMTS	0.0146	0.0744*	0.0232	0.0385*
OTS	0.0143*	0.0725*	0.0224*	0.0384*
TexLexAn	0.0134*	0.0709*	0.0212*	0.0379*

Table XXII. English News Collections: Evaluation by Means of the the ROUGE Toolkit
Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}minsup = 0.8\%$, $K = 6$, stopword elimination and stemming) and the other approaches are starred.

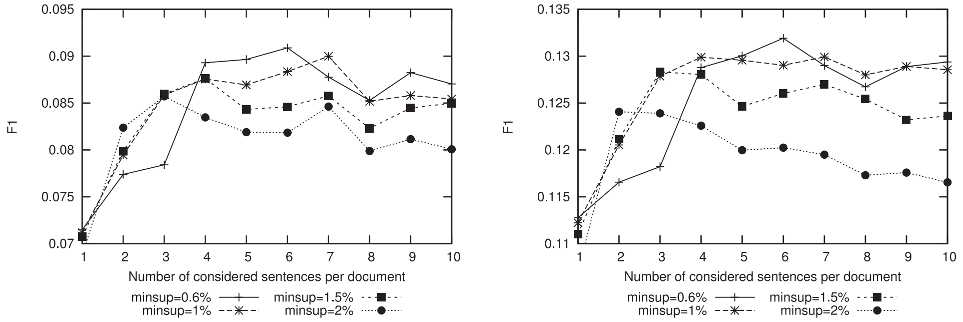
	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
MWI-Sum	**0.0185**	**0.0998**	**0.0296**	**0.0322**	**0.1730**	**0.0518**
ICSIsumm	0.0179	0.0985	0.0289	0.0298*	0.1586*	0.0478*
ItemSum	0.0151*	0.0919*	0.0238*	0.0269*	0.1628*	0.0427*
AMTS	0.0146	0.0744*	0.0232	0.0309	0.1536	0.0488
OTS	0.0143*	0.0725*	0.0224*	0.0272*	0.1455*	0.0434*
TexLexAn	0.0134*	0.0709*	0.0212*	0.0289*	0.1505*	0.0459*

Table XXIII. Italian News Collections: Evaluation by Means of JRouge and AutoSummENG
Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}minsup = 3\%$, $K = 6$, stopword elimination) and the other approaches are starred.

	JRouge ROUGE-2			AutoSummENG
Summarizer	R	Pr	F1	Avg. Similarity
MWI-Sum	**0.0206**	**0.0696**	**0.0300**	**0.0593**
AMTS	0.0178	0.0641	0.0264	0.0515
OTS	0.0171	0.0596	0.0253	0.0505*
ICSIsumm	0.0152	0.0549	0.0223	0.0546
TexLexAn	0.0145	0.0509	0.0215	0.0556
ItemSum	0.0103*	0.0435*	0.0159*	0.0489

Table XXIV. Italian News Collections: Evaluation by Means of the the ROUGE Toolkit
Statistically relevant differences in the comparison between MWI-Sum ($\mathcal{W}minsup = 3\%$, $K = 6$, stopword elimination) and the other approaches are starred.

	ROUGE toolkit					
	ROUGE-2			ROUGE-SU4		
Summarizer	R	Pr	F1	R	Pr	F1
MWI-Sum	**0.0206**	**0.0697**	**0.0301**	**0.0368**	**0.1342**	**0.0548**
AMTS	0.0174	0.0624	0.0258	0.0342	0.1233	0.0508
TexLexAn	0.0171	0.0598	0.0253	0.0336	0.1212	0.0498
OTS	0.0149	0.0524	0.0221	0.0307	0.1106	0.0456
ICSIsumm	0.0150	0.0538	0.0220	0.0329	0.1202	0.0487
ItemSum	0.0102*	0.0429	0.0158*	0.0297*	0.1159	0.0449*



(a) Rouge-2

(b) Rouge-SU4

Fig. 2. F1 measure: impact of the number of considered sentences per document parameter.

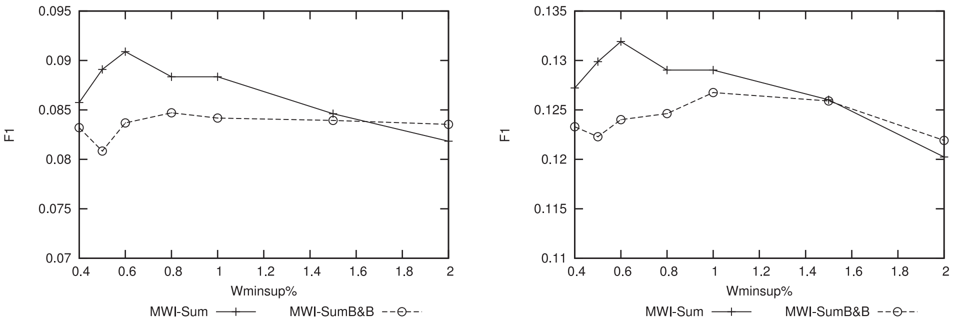
non–English-written documents, we experimentally compared the performance of the MWI-Sum summarizer with that of (i) ItemSum and (ii) a slightly modified version of the MWI-Sum summarizer, hereafter denoted as MWI-SumBaseline, in which document terms have all the same relevance within the document collections, i.e., all item weights are set to 1. While ItemSum performs a different summarization process (e.g., a different sentence filtering step, tf-idf-based instead of tf-df term evaluation), MWI-SumBaseline relies on the same summarization steps as MWI-Sum.

The experimental results show that for all languages, except for Hindi, MWI-Sum performs better than both MWI-SumBaseline and ItemSum on the TAC'11 collections in terms of both JRouge-2 F1-measure and AutoSummENG average similarity. On English-written documents, the performance improvement is limited (JRouge-2 F1-measure +0.0045), whereas on most of the other languages, the performance gap becomes larger (e.g., JRouge-2 F1-measure +0.0153 on Greek, +0.0247 on French). On Hindi-written documents, MWI-Sum performs better than ItemSum (JRouge-2 F1-measure +0.023) and slightly worse than MWI-SumBaseline. However, the gap between MWI-SumBaseline and MWI-Sum is small (JRouge-2 F1-measure −0.001). In summary, item weights are particularly effective for summarizing non–English-written documents.

4.6. Parameter Analysis

We analyzed the impact of MWI-Sum parameters and features on its summarization performance. Specifically, we analyzed the impact of (i) the cardinality K of the set of top sentences selected from each document during the preprocessing step, (ii) the minimum weighted support threshold $\mathcal{W}minsup$ enforced during the frequent weighted itemset mining step, (iii) the use of the greedy strategy to perform the coverage of the set of extracted itemsets, and (iv) the use of maximal and closed itemsets instead of traditional frequent itemsets. For the experimental evaluation, we tested our summarizer on the DUC'04 collection by using the experimental setting described in Section 4.3. For the sake of brevity, we reported detailed results only in terms of ROUGE-2 F1 and ROUGE-SU4 F1.

As shown in Figures 2 and 3, the weighted support threshold may significantly affect the performance of the MWI-Sum summarizer, in particular in terms of ROUGE-SU4. In the performed experiments, we have tested various support threshold values between 0.6% and 2%. When high support thresholds (e.g., 2%) are enforced, many informative itemsets are discarded. Hence, the itemset-based model becomes too general to yield good summarization performance. On the other hand, when low support thresholds are enforced (e.g., 0.6%), a larger set of itemsets is mined and a more complete



(a) Rouge-2

(b) Rouge-SU4

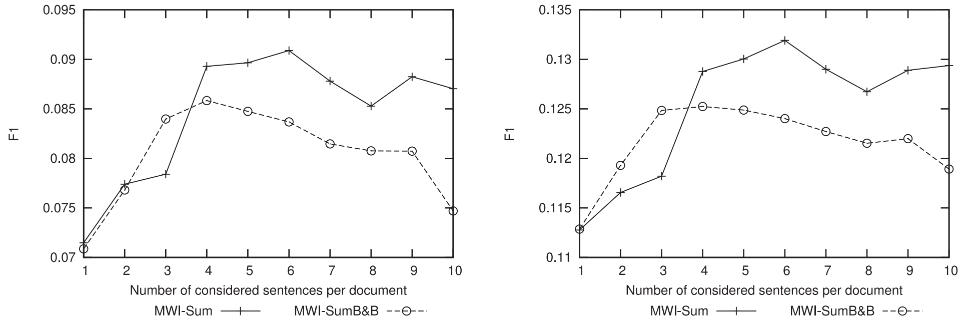
Fig. 3. Comparison between MWI-Sum and MWI-Sum$_{B\&B}$ in terms of F1 measure when varying the weighted support threshold $\mathcal{W}$minsup ($K = 6$).

and accurate itemset-based model is generated. Thus, MWI-Sum achieves, on average, better performance. The best support threshold setting actually depends on the analyzed data distribution and language. However, since for support threshold values below or equal to 1%, MWI-Sum on average achieves fairly good results on the analyzed multilingual collections, we recommend setting values in this range.

In Figures 2(a) and 2(b), we plot the variation of the ROUGE-2 and ROUGE-SU4 F1-measure when varying the value of K. Different curves are plotted for different weighted support threshold values. The number of considered sentences per document affects the summarization performance only when relatively low values (e.g., less than 3) or high values (e.g., higher than 10) are considered. When a very limited number of sentences (e.g., $K = 2$) is selected, most of the relevant knowledge hidden in the document collection is not covered by any selected sentence. Thus, the generated summaries are not highly informative. For $K \geq 3$, the performance trend is relatively stable when low support thresholds (e.g., 0.6%) are enforced. The performance becomes more sensitive to K for higher support thresholds (e.g., 2%), which also yield low quality summaries. A similar trend is shown on the TAC'11 multilingual collections.

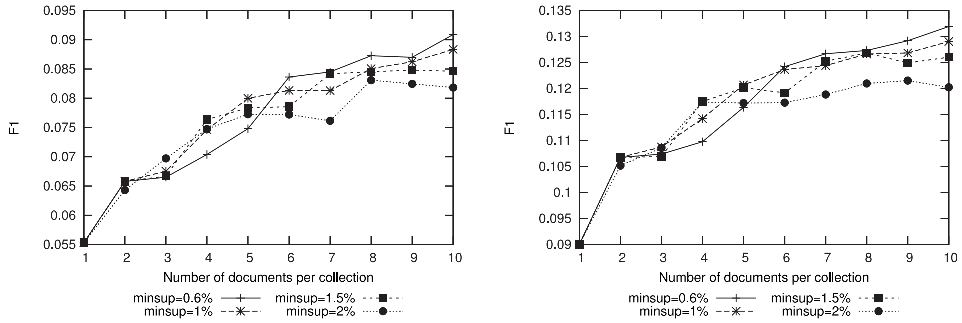
MWI-Sum exploits a greedy strategy to solve the set covering optimization problem. Generally speaking, it produces an approximated solution to the problem of selecting the set of sentences that best covers the itemset-based model. Since the set covering problem is a min-max problem, it may be converted into a linear programming problem and addressed using combinatorial optimization strategies. To evaluate the impact of the greedy algorithm on the summarization performance, we developed MWI-Sum$_{B\&B}$, a slightly modified version of the MWI-Sum system that exploits a branch-and-bound algorithm [Ralphs and Guzelsoy 2006] to perform the set covering task. Figures 3 and 4 compare the values of the ROUGE-2 and ROUGE-SU4 F1 measures provided by MWI-Sum and MWI-Sum$_{B\&B}$ by varying the weighted support threshold $\mathcal{W}$minsup and the number of selected sentences per document K, respectively. MWI-Sum allows achieving higher results than MWI-Sum$_{B\&B}$ in terms of ROUGE-2 and ROUGE-SU4 F1-measure because of the lower sensitivity of the generated itemset-based model to noise and data overfitting. Furthermore, MWI-Sum requires, on average, an execution time at least 10% lower than MWI-Sum$_{B\&B}$ for every considered parameter setting.

To experimentally evaluate the effectiveness of the tf-df statistics for summarizing documents ranging over the same topic, we implemented a slightly modified MWI-Sum version that integrates the tf-idf statistics instead of the tf-df one. The results of the experiments, not reported here for the sake of brevity, show that the use of the tf-df statistics significantly improves the summarization performance (e.g., for the best



(a) Rouge-2 (b) Rouge-SU4

Fig. 4. Comparison between MWI-Sum and MWI-Sum$_{B\&B}$ in terms of F1-measure when varying the number of considered sentences per document ($\mathcal{W}$minsup=0.6%).



(a) Rouge-2 (b) Rouge-SU4

Fig. 5. F1 measure: impact of the number of documents per collection ($K = 6$).

configuration 0.1319 with respect to 0.0909 for the ROUGE-SU4 F1-measure on the DUC’04 collections).

We also considered maximal and closed frequent itemsets instead of frequent itemsets to drive the summarization process. Specifically, we integrated in MWI-Sum two traditional closed [Pasquier et al. 1999] and maximal [Roberto and Bayardo 1998] itemset mining algorithms. The experimental results show that the use of maximal and closed frequent itemsets, on average, lowers MWI-Sum performance, because some valuable correlations among terms are not included in the closed/maximal itemset-based model (e.g., ROUGE-SU4 F1-measure 0.1300 with closed for the best configuration on the DUC’04 collections).

4.7. Analysis of the Collection Size

All the analyzed collections consist of at least 10 documents each. According to the DUC’04 and TAC’11 contest policies, one summary per collection is generated by each summarizer. Since the number of documents per group may significantly affect the result of the summarization process, we experimentally analyzed its impact on the DUC’04 collections. Specifically, we performed 10 rounds of experiments by considering groups of documents of size ranging from 1 to 10. The groups with less than 10 documents were generated by sampling the original DUC’04 collection. The achieved results are reported in Figure 5. As expected, when few documents per group are available (less than four) the performance results are significantly worse than the best ones, whereas averagely high ROUGE scores were achieved on groups consisting of six documents or more, because the generated itemset-based model becomes more reliable. Similar trends were obtained by testing the other approaches.

5. CONCLUSIONS AND FUTURE WORK

M W I-Sum is a novel multilingual summarizer that exploits an itemset-based model, composed of frequent weighted itemsets, to summarize multidocument collections ranging over the same topic. The proposed summarizer does not rely on complex semantics-based models (e.g., ontologies or taxonomies). The use of language-dependent text analyses is limited to stopword elimination and stemming. Hence, MWI-Sum is easily portable to languages other than English.

Experiments, performed on a large variety of benchmark and real-life datasets, show the effectiveness of the proposed summarizer in coping with documents written in different languages, European and not. MWI-Sum is shown to perform better than state-of-art summarizers on multilingual collections, while it is competitive with language-specific summarizers on English-written collections. The experimental results show that exploiting item weights during itemset mining is particularly effective for summarizing non–English-written documents.

Even though most of the benchmark multilingual documents are news ranging over the same topic, we envision examples of different application domains where the MWI-Sum summarizer can find application. For example, in e-learning systems, learners can share personal notes on the same topic on the e-learning platform. Generating per-topic note summaries is appealing because they can be used as additional material for teaching and revision [Baralis et al. 2015]. On blogs, mailing lists, and forums, social network users commonly post medium-length texts ranging over specific topics (e.g., financial instruments, football matches, cars). This textual content can be analyzed to pinpoint relevant knowledge. For example, to drive private investor's decisions, messages posted by expert traders on specific assets (e.g., bonds, shares) can be analyzed. Finally, based on the context under analysis, the summarization can be driven by user skills and/or tailored to user-specific needs. For example, in an e-learning context documents uploaded by highly skilled users can be deemed as more significant for summarization purposes than those uploaded by lowly skilled ones.

APPENDIX

In this appendix, we formally prove Property 3.3 and Property 3.4 introduced in Section 3.

PROPERTY 3.3 [EQUIVALENCE PROPERTY]. *Let T be a weighted transactional dataset and ET its equally weighted representation. The weighted supports of an arbitrary weighted itemset I in T and ET are equal.*

PROOF. Let $tr = \{\langle w_1, td_1\rangle, \langle w_2, td_2\rangle, \ldots, \langle w_m, td_m\rangle\} \in T$ be a weighted transaction matched by I and $ET_{tr} = \{et_1, \ldots, et_k\}$ its equally weighted transactional representation. The proof is divided into three steps: (i) For every term $w_i | w_i \in tr$, its absolute support in ET_{tr} is proved to be equal to td_i, i.e., w_i's weight in tr. (ii) The normalization term is computed by proving the following relationship between the item weights in ET_{tr} and T: $\sum_{et_p \in ET_{tr}} \max_{u|w_u \in et_p} tde_u = \max_{i|w_i \in tr} td_i$, where tde_u is the weight of the u-th item in $et_p \in ET_{tr}$. (iii) The final assertion $\mathcal{W}sup(I, ET) = \mathcal{W}sup(I, T)$ is proved.

Step (i). Let $\langle w_i, td_i\rangle$ be the p-th item in increasing weight order of the weighted transaction tr and $td_{p^*} \in \overline{W}$ the p-th lowest term weight in tr. From Definition 3.2, the following equality holds: $td_i = td_{p^*}$. Hence, term w_i with weight td_i is contained in each of the first p equivalent transactions $et_1, \ldots, et_p$ in ET_{tr}. Therefore, the absolute weighted support of w_i in ET_{tr} can be computed as follows:

$$\sum_{1\le n\le p} tde_n = tde_1 + \sum_{2\le n\le p} tde_n = td_{1^*} + \sum_{2\le n\le p} (td_{n^*} - td_{n-1^*}) = td_{p^*} = td_i$$

Step (ii). Since each item in et_p has weight tde_p, from Definition 3.2 it follows that:

$$\sum_{et_p\in ET_{tr}} \max_{u|w_u\in et_p} tde_u = \sum_{et_p\in ET_{tr}} tde_p = \sum_{1\le n\le k} tde_n = td_{k^*} = \max_{i|w_i\in tr} td_i$$

Step (iii). Let $matched = \{w_i \in tr \mid w_i \in I\}$ be the set of items in tr matched by I and $i_L = \langle w_L, td_L\rangle$ the least weighted item in *matched*. The matching weight of I in tr is given by $\mathcal{W}(I, tr) = td_L$. Furthermore, since i_L is the least weighted item in *matched*, from Definition 3.2, it follows that all the transactions in ET_{tr} that contain i_L also include every other item in *matched*. Let $matchedET_{tr}$ be the corresponding subset of equivalent transactions that match I. Since $matchedET_{tr} \subseteq ET_{tr}$ includes all the equivalent transactions et_p that contain i_L, then, from Step (i), it follows that the absolute weighted support of I in ET_{tr} is equal to the matching weight of i_L in tr, i.e., $\mathcal{W}(I{,}ET_{tr}) = td_L = \mathcal{W}(I, tr)$. Note that the contribution of every equally weighted transaction $et_j \in ET_{tr} \mid I \nsubseteq et_j$ to the absolute weighted support of I in ET_{tr} is zero.

The absolute weighted support of I in the equivalent dataset ET is given by the summation of the absolute weighted supports of I in all the equivalent sets ET_{tr} that are generated from every $tr \in T$. Hence, its expression can be rewritten as follows:

$$\mathcal{W}(I, ET) = \sum_{ET_{tr}|tr\in T} \mathcal{W}(I, ET_{tr}) = \sum_{tr\in T} \mathcal{W}(I, tr) = \mathcal{W}(I, T)$$

Combining the above equality with the one stated at Step (ii), it follows that:

$$\mathcal{W}sup(I, ET) = \frac{\mathcal{W}(I, ET)}{\sum_{et_v\in ET} \max_{u|w_u\in et_v} tde_u}$$

$$= \frac{\mathcal{W}(I, ET)}{\sum_{tr\in T}\sum_{et_p\in ET_{tr}} \max_{u|w_u\in et_p} tde_u} = \frac{\mathcal{W}(I, T)}{\sum_{tr\in T} \max_{i|w_i\in tr} td_i}$$

$$= \mathcal{W}sup(I, T) \quad \square$$

PROPERTY 3.4. *Let D be a document collection and M the* MWI-S*um itemset-based model that is built on D by enforcing a minimum weighted support threshold $\mathcal{W}minsup > 0$. The summary $\mathcal{SU}$ generated by* MWI-S*um fully covers M.*

PROOF. Let D be a document collection and T its weighted transactional version. Let $I \in M$ be an arbitrary frequent weighted itemset mined from T. Since $\mathcal{W}minsup > 0$ and I is mined from T, then it follows that $\mathcal{W}sup(I, T) > 0$. Hence, there exists at least one transaction $tr_j \in T$ such that the matching weight of I with respect to T is greater than 0. It follows that there exists at least one sentence $s_j \in D$, associated with the transaction tr_j, such that s_j covers I.

The greedy sentence selection approach iterates until at least one itemset I is uncovered (i.e., SC^* contains at least one zero). Since for each $I \in M$ there exists at least one sentence $s_j \in D$ covering it, it follows that the summary $\mathcal{SU}$ contains at least one sentence covering I, i.e., $\mathcal{SU}$ contains s_j or another sentence covering I and characterized by a sentence coverage higher than those of s_j. $\square$

REFERENCES

John Atkinson and Ricardo Munoz. 2013. Rhetorics-based multi-document summarization. *Expert Syst. Appl.* 40, 11 (2013), 4346–4352. DOI:http://dx.doi.org/10.1016/j.eswa.2013.01.017

Elena Baralis, Luca Cagliero, and Laura Farinetti. 2015. Generation and evaluation of summaries of academic teaching materials. In *Proceedings of the 39th Annual IEEE Computer Software and Applications Conference (COMPSAC'15)*. 881–886. DOI:http://dx.doi.org/10.1109/COMPSAC.2015.15

Elena Baralis, Luca Cagliero, Saima Jabeen, and Alessandro Fiori. 2012. Multi-document summarization exploiting frequent itemsets. In *Proceedings of the ACM Symposium on Applied Computing (SAC'12)*. 782–786. DOI:http://dx.doi.org/10.1145/2245276.2245427

Elena Baralis, Luca Cagliero, Saima Jabeen, Alessandro Fiori, and Sajid Shah. 2013a. Multi-document summarization based on the Yago ontology. *Expert Syst. Appl.* 40, 17 (2013), 6976–6984.

Elena Baralis, Luca Cagliero, Naeem A. Mahoto, and Alessandro Fiori. 2013b. GraphSum: Discovering correlations among multiple terms for graph-based summarization. *Inf. Sci.* 249 (2013), 96–109.

Elena Maria Baralis, Luca Cagliero, Alessandro Fiori, and Saima Jabeen. 2011. PatTexSum: A pattern-based text summarizer. In *Proceedings of the Mining Complex Patterns Workshop*. 18–29. Retrieved from http://porto.polito.it/2460874/.

Regina Barzilay and Michael Elhadad. 1997. Using lexical chains for text summarization. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*. 10–17.

S. Bird, E. Klein, and E. Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Sergey Brin and Lawrence Page. 1998. The anatomy of a large-scale hypertextual Web search engine. In *Proceedings of the 7th International Conference on World Wide Web* 7. 107–117.

Luca Cagliero and Paolo Garza. 2014. Infrequent weighted itemset mining using frequent pattern growth. *IEEE Trans. Knowl. Data Eng.* 26, 4 (2014), 903–915. DOI:http://dx.doi.org/10.1109/TKDE.2013.69

Giuseppe Carenini, Raymond T. Ng, and Xiaodong Zhou. 2007. Summarizing email conversations with clue words. In *World Wide Web Conference Series*. 91–100.

Wesley T. Chuang and Jihoon Yang. 2000. Extracting sentence segments for text summarization: A machine learning approach. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'00)*. ACM, New York, NY, 152–159. DOI:http://dx.doi.org/10.1145/345508.345566

John Conroy, Judith Schlesinger, Jeff Kubina, Peter Rankel, and Dianne OLeary. 2011. CLASSY 2011 at TAC: Guided and multi-lingual summaries and evaluation metrics. In *TAC'11: Proceedings of the the 2011 Text Analysis Conference (TAC'11)*.

John M. Conroy, Jade Goldstein, Judith D. Schlesinger, and Dianne P. OLeary. 2004. Left-brain/right-brain multi-document summarization. In *DUC 2004 Conference Proceedings*.

Wordnet Lexical Database. 2012. Homepage. Available at http://wordnet.princeton.edu.

Thomas G. Dietterich. 1998. Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation* 10, 7 (1998).

Document Understanding Conference. 2004. HTL/NAACL Workshop on Text Summarization. http://duc.nist.gov/pubs.html#2004.

Mark Dredze, Hanna M. Wallach, Danny Puller, and Fernando Pereira. 2008. Generating summary keywords for emails using topics. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI'08)*. ACM, New York, NY, 199–206. DOI:http://dx.doi.org/10.1145/1378773.1378800

Elena Filatova. 2004. A formal model for information selection in multi-sentence text extraction. In *Proceedings of the International Conference on Computational Linguistics (COLING'04)*. 397–403.

Garcia Lus Fernando Fortes, de Lima Jos Valdeni, Stanley Loh, and Jos Palazzo Moreira de Oliveira. 2006. Using ontological modeling in a context-aware summarization system to adapt text for mobile devices. In *Active Conceptual Modeling of Learning (Lecture Notes in Computer Science)*, Peter P. Chen and Leah Y. Wong (Eds.), Vol. 4512. Springer, 144–154.

George Giannakopoulos, Mahmoud El-Haj, Benoit Favre, Marina Litvak, Josef Steinberger, and Vasudeva Varma. 2011. TAC2011 MultiLing Pilot Overview. In *Proceedings of the TAC 2011 Workshop*. NIST, Gaithersburg, MD, Retreived from http://users.iit.demokritos.gr/∼ggianna/Publications/MultiLingOverview.pdf.

George Giannakopoulos and Vangelis Karkaletsis. 2011. AutoSummENG and MeMoG in evaluating guided summaries. In *Proceedings of the TAC 2011 Workshop*. NIST. Retrieved from http://users.iit.demokritos.gr/∼ggianna/Publications/TAC2011-AESOPSystemPresentation.pdf.

Dan Gillick, Benoit Favre, and Dilek Hakkani-Tur. 2008. The ICSI summarization system at TAC 2008. In *Proceedings of the Text Analysis Conference (TAC'08)*.

Dan Gillick, Benoit Favre, Dilek Hakkani-Tur, Bernd Bohnet, Yang Liu, and Shasha Xie. 2009. The ICSI/TUD summarization system at TAC 2009. In *Proceedings of the Text Analysis Conference (TAC'09)*.

Gösta Grahne and Jianfei Zhu. 2003. Efficiently using prefix-trees in mining frequent itemsets. In *Proceedings of the Workshop on Frequent Itemset Mining Implementations, FIMI'03 (CEUR-WS)*, Bart Goethals and Mohammed J. Zaki (Eds.), Vol. 90.

Oskar Gross, Antoine Doucet, and Hannu Toivonen. 2014. Document summarization based on word associations. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'14)*. ACM, New York, NY, 1023–1026. DOI:http://dx.doi.org/10.1145/2600428.2609500

Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. 2003. Clustering data streams: Theory and practice. *IEEE Trans. on Knowl. and Data Eng.* 15, 3 (March 2003), 515–528. http://dx.doi.org/10.1109/TKDE.2003.1198387.

Jiawei Han, Hong Cheng, Dong Xin, and Xifeng Yan. 2007. Frequent pattern mining: Current status and future directions. *Data Min. Knowl. Discov.* 15, 1 (2007), 55–86.

Jiawei Han, Jain Pei, and Yiwen Yin. 2000. Mining frequent patterns without candidate generation. *In SIGMOD'00.*

Leonhard Hennig, Winfried Umbrath, and Robert Wetzker. 2008. An ontology-based approach to text summarization. In *Web Intelligence/IAT Workshops*. IEEE, 291–294.

Jiri Hynek and Karel Jezek. 2003. Practical approach to automatic text summarization. In *ELPUB*.

Szymon Jaroszewicz and Dan A. Simovici. 2004. Interestingness of frequent itemsets using Bayesian networks as background knowledge. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 178–186.

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

A. Kogilavani and B. Balasubramanie. 2009. Ontology enhanced clustering based summarization of medical documents. *Int. J. Recent Trends Engin.* 1, 1 (2009).

Eugene Krapivin, Mark Last, and Marina Litvak. 2014. JRouge - Java ROUGE Implementation. Retrieved from https://bitbucket.org/nocgod/jrouge/wiki/Home/.

Lei Li, Dingding Wang, Chao Shen, and Tao Li. 2010. Ontology-enriched multi-document summarization in disaster management. In *SIGIR*, Fabio Crestani, Stphane Marchand-Maillet, Hsin-Hsi Chen, Efthimis N. Efthimiadis, and Jacques Savoy (Eds.). ACM, 819–820.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using N-gram co-occurrence statistics. In *Proceedings of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*. 71–78.

Hui Lin and Jeff Bilmes. 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT'11)*. Association for Computational Linguistics, Stroudsburg, PA, 510–520.

Edward Loper and Steven Bird. 2002. NLTK: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1 (ETMTNLP'02)*. Association for Computational Linguistics, Stroudsburg, PA, 63–70. DOI:http://dx.doi.org/10.3115/1118108.1118117

Michael Mampaey, Nikolaj Tatti, and Jilles Vreeken. 2011. Tell me what I need to know: Succinctly summarizing data with itemsets. In *Proceedings of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Michael McCandless, Erik Hatcher, and Otis Gospodnetic. 2010. *Lucene in Action, Second Edition: Covers Apache Lucene 3.0*. Manning Publications Co., Greenwich, CT.

Jade Goldstein Vibhu Mittal, Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. 2000. Multi-document summarization by sentence extraction. In *Proceedings of the ANLP/NAACL Workshop on Automatic Summarization*. 40–48.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Comput. Linguist.* 17, 1 (March 1991), 21–48.

Nicolas Pasquier, Yves Bastide, Rafik Taouil, and Lotfi Lakhal. 1999. Discovering frequent closed itemsets for association rules. In *Proceedings of the 7th International Conference on Database Theory (ICDT'99)*. Springer-Verlag, London, UK, 398–416.

Chen Ping and Verma Rakesh M. 2006. A query-based medical information summarization system using ontology knowledge. In *CBMS*. IEEE Computer Society, 37–42.

Mohsen Pourvali and Mohammad Saniee Abadeh. 2012. Automated text summarization base on lexicales chain and graph using of WordNet and Wikipedia knowledge base. *CoRR* abs/1203.3586 (2012).

Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.* 22 (2004), 2004.

Dragomir R. Radev, Hongyan Jing, Malgorzata Stys, and Daniel Tam. 2004. Centroid-based summarization of multiple documents. *Inf. Process. Manag.* 40, 6 (2004), 919–938.

T. Ralphs and M. Guzelsoy. 2006. The SYMPHONY callable library for mixed integer programming. *The Next Wave in Computing, Optimization, and Decision Technologies* 29 (2006), 61–76. Software available at http://http://www.coin-or.org/SYMPHONY.

J. Roberto and J.r. Bayardo. 1998. Efficiently mining long patterns from databases. In *SIGMOD 1998*, Laura M. Haas and Ashutosh Tiwary (Eds.). 85–93.

N. Rotem. 2011. Open Text Summarizer (OTS). Retrieved from http://libots.sourceforge.net/.

Josef Steinberger, Mijail Kabadjov, Ralf Steinberger, Hristo Tanev, Marco Turchi, and Vanni Zavarella. 2011. JRC's participation at TAC 2011: Guided and multilingual summarization tasks. In *Proceedings of the 2011 Text Analysis Conference (TAC'11)*.

Ke Sun and Fengshan Bai. 2008. Mining weighted association rules without preassigned weights. *IEEE Trans. Knowl. Data Eng.* 20, 4 (April 2008), 489–495. DOI:http://dx.doi.org/10.1109/TKDE.2007.190723

Hiroya Takamura and Manabu Okumura. 2009. Text summarization model based on the budgeted median problem. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management*. 1589–1592.

Pang-Ning Tan, Vipin Kumar, and Jaideep Srivastava. 2002. Selecting the right interestingness measure for association patterns. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'02)*. 32–41.

Nikolaj Tatti. 2010. Probably the best itemsets. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 293–302.

TexLexAn. 2011. TexLexAn: An Open-Source Text Summarizer. Retrieved from http://texlexan.sourceforge.net/.

Text Analysis Conference. 2011. NIST Text Analysis Conference Summarization Track. Retrieved from http://www.nist.gov/tac/2011/Summarization.

K. S. Thakkar, R. V. Dharaskar, and M. B. Chandak. 2010. Graph-based algorithms for text summarization. In *Proceedings of the 2010 3rd International Conference on Emerging Trends in Engineering and Technology (ICETET)*. 516–519. DOI:http://dx.doi.org/10.1109/ICETET.2010.104

Merijn van Erp and Lambert Schomaker. 2000. Variants of the borda count method for combining ranked classifier hypotheses. In *Proceedings of the 7th International Workshop on Frontiers in Handwriting Recognition*. 443–452.

Xiaojun Wan and Jianwu Yang. 2006. Improved affinity graph based multi-document summarization. In *Proceedings of HLT-NAACL, Companion Volume: Short Papers*. 181–184.

Dingding Wang and Tao Li. 2010. Document update summarization using incremental hierarchical clustering. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. 279–288.

Dingding Wang, Shenghuo Zhu, Tao Li, Yun Chi, and Yihong Gong. 2011. Integrating document clustering and multidocument summarization. *ACM Trans. Knowl. Discov. Data* 5, 3, (August 2011), Article 14, 26 pages. DOI:http://dx.doi.org/10.1145/1993077.1993078

Dingding Wang, Shenghuo Zhu, Tao Li, and Yihong Gong. 2013. Comparative document summarization via discriminative sentence selection. *ACM Trans. Knowl. Discov. Data* 7, 1, Article 2 (March 2013), 18 pages. DOI:http://dx.doi.org/10.1145/2435209.2435211

Wei Wang, Jiong Yang, and Philip S. Yu. 2000. Efficient mining of weighted association rules (WAR). In *Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 270–274.

Chia-Wei Wu and Chao-Lin Liu. 2003. Ontology-based text summarization for business news articles. In *Computers and Their Applications*, Narayan C. Debnath (Ed.). ISCA, 389–392.

Xindong Wu, Gong-Qing Wu, Fei Xie, Zhu Zhu, and Xue-Gang Hu. 2010. News filtering and summarization on the web. *IEEE Intell. Syst.* 25, 5 (Sept. 2010), 68–76. DOI:http://dx.doi.org/10.1109/MIS.2010.11

Zi Yang, Keke Cai, Jie Tang, Li Zhang, Zhong Su, and Juanzi Li. 2011. Social context summarization. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11)*. ACM, New York, NY, 255–264. DOI:http://dx.doi.org/10.1145/2009916.2009954

Junyan Zhu, Can Wang, Xiaofei He, Jiajun Bu, Chun Chen, Shujie Shang, Mingcheng Qu, and Gang Lu. 2009. Tag-oriented document summarization. In *Proceedings of the 18th International Conference on World Wide Web (WWW'09)*. ACM, New York, NY, 1195–1196. DOI:http://dx.doi.org/10.1145/1526709.1526925

Received June 2014; revised June 2015; accepted July 2015