PatTexSum : A pattern-based text summarizer

Elena Baralis, Luca Cagliero, Alessandro Fiori, and Saima Jabeen

Abstract In the last decade the growth of the Internet has made a huge amount of textual documents available in the electronic form. Text summarization is commonly based on clustering or graph-based methods and usually considers the bag-of-word sentence representation. Frequent itemset mining is a widely exploratory technique to discover relevant correlations among data. The well-established application of frequent itemsets to large transactional datasets prompts their usage in the context of document summarization as well.

This paper proposes a novel multi-document summarizer, namely PatTexSum (Pattern-based Text SUMmarizer), that is mainly based on a pattern-based model, i.e., a model composed of frequent itemsets. Unlike previously proposed approaches, PatTexSum selects most representative and not redundant sentences to include in the summary by considering both (i) the most informative and non-redundant itemsets extracted from document collections tailored to the transactional data format, and (ii) a sentence score, based on the tf-idf statistics. Experiments conducted on a collection of real news articles show the effectiveness of the proposed approach.

Key words: Text mining, Document summarization, Frequent itemset mining

1 Introduction

From the birth of the Internet on, analysts may progressively access and analyze larger data collections. Since the large majority of the information is

e-mail: {elena.baralis,luca.cagliero,alessandro.fiori,saima.jabeen}@polito.it
Affiliation: Politecnico di Torino. Corso Duca degli Abruzzi, 24 10129 Torino, Italy. Tel: +390110907194 Fax:+390110907099

available in textual form, a challenging task is to convey the most relevant information provided by textual documents into short and concise summaries.

Many document summarization approaches have been proposed in literature. Most of them select the most representative sentences to include in the summaries by means of the following approaches: (i) clustering (e.g., [13, 20]), (ii) graph-based methods (e.g., [12]), and (iii) linear programming (e.g., [15]). Clustering-based approaches exploit clustering algorithms to group sentences and select representatives among each group. For instance, MEAD [13] evaluates the similarity between the document sentences and the centroids and selects, similarly to [6], the most relevant sentences among each document cluster based on the tf-idf statistical measure [16]. Differently, in [20] an incremental hierarchical clustering algorithm is exploited to update summaries over time. The graph-based approaches try to represent correlations among sentences by means of a graph-based model. According to this model, sentences are represented by graph nodes, while the edges weigh the strength of the correlation between couples of sentences. The most representative sentences are selected according to graph-based indexing strategies. For instance, [12] proposes to rank sentences based on the eigenvector centrality computed by means of the well-known PageRank algorithm [5]. Finally, the linear programming methods identify the most representative sentences by maximizing ad-hoc object functions. For instance, in [15] the authors formalized the extractive summarization task as a maximum coverage problem with the Knapsack constraints based on the the bag-of word sentence representation and enforce additional constraints based on sentence relevance within each document. Most the aforementioned approaches rely on the bag-of-word sentence representation and make use of well-founded statistical measures (e.g., the tf-idf measure [16]).

Frequent itemset mining is a widely exploratory technique, first introduced in [1] in the context of market basket analysis, to discover correlations that frequently occur in the analyzed data. A number of approaches focus on discovering frequent itemsets from transactional data and then selecting their most informative yet non-redundant subset by means of postpruning. To address this issue, static approaches (e.g., [4, 8]) compare the observed frequency (i.e., the support) of each itemset in the source transactional data against some null hypotheses (i.e., their expected frequency). Differently, dynamic approaches (e.g., [9, 18]) make often use of the maximum entropy model to take previously selected patterns into account and, thus, reduce model redundancy. Although the discovery and selection of valuable frequent itemsets from transactional data is well-established, to the best of our knowledge their usage in document summarization has never been investigated yet.

PatTexSum (Pattern-based Text SUMmarizer) is a novel multi-document summarization approach that exploits a pattern-based model to select the most representative and not redundant sentences belonging to the document collection. It focuses on combining the effectiveness of pattern-based models, composed of highly informative and non-redundant itemsets, to represent correlations among data with the discriminating power of a sentence evaluation measure, based the tf-idf statistics. Pattern-based model generation focuses on extracting and selecting valuable frequent itemsets from a transactional representation of the document collection. To this aim, an efficient and effective approach, recently proposed in [11] in the context of transactional data, is adopted. [11] succinctly summarizes transactional data by adopting an heuristics to solve the maximum entropy model that allows on-the-fly evaluating itemsets during their extraction. This feature makes this approach particularly appealing for its application in text summarization. To effectively discriminate among sentences, an evaluation score, computed from their bag-of-word representation and based on the well-founded tf-idf statistic [16], is also considered. PatTexSum combines the information discovered from both transactional and bag-of-word data representations and adopts an effective greedy approach, first proposed in [2], to solve the problem of selecting sentences that cover at best the pattern-based model.

To evaluate the PatTexSum performance a suite of experiments on a collection of news articles has been performed. Results, reported in Section 3, show that PatTexSum significantly outperforms mostly used previous summarizers in terms of precision, recall, and F-measure.

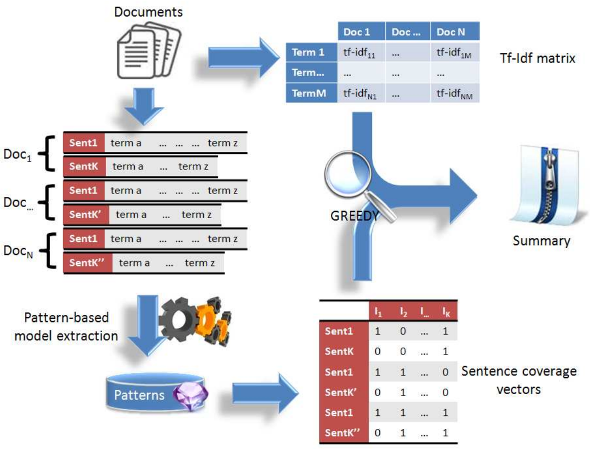
This paper is organized as follows. Section 2 presents the proposed method and thoroughly describes its main steps. Section 3 assesses the effectiveness of the PatTexSum framework in summarizing textual documents, while Section 4 draws conclusions and presents future developments of this work.

2 The PatTexSum method

PatTexSum focuses on summarizing collections of textual documents by exploiting a two-way data representation. Pattern-based model generation relies on a transactional representation of the document sentences, while the relevance score evaluation, based on the tf-idf statistic, is based on the bag-of-word sentence representation. A greedy approach is used to effectively combine knowledge discovered from both data representations and select most representative sentences to include in the summary. Figure 1 shows the main steps behind the proposed approach, which will be thoroughly described in the following.

2.1 Document representation

PatTexSum exploits two different document/sentence representations: (i) the traditional bag-of-word (BOW) representation and (ii) the transactional data format. The raw document content is first preprocessed to make it



	Doc 1	Doc ...	Doc N
Term 1	tf-idf$_{11}$	...	tf-idf$_{1M}$
Term...	...	...	...
TermM	tf-idf$_{N1}$	...	tf-idf$_{NM}$

Sent1	term a	...	...	...	term z
SentK	term a	...	term z		
Sent1	term a	...	...	...	term z
SentK'	term a	...	term z		
Sent1	term a	...	...	...	term z
SentK''	term a	...	term z		

	I_1	I_2	$I_{\_}$	I_k
Sent1	1	0	...	1
SentK	0	0	...	1
Sent1	1	1	...	0
SentK'	0	1	...	0
Sent1	1	1	...	1
SentK''	0	1	...	1

Fig. 1 The PatTexSum method

suitable for the data mining and knowledge discovery process. Stopwords, numbers, and website URLs are removed to avoid noisy information, while the Wordnet stemming algorithm [3] is applied to reduce document words to their base or root form (i.e., the stem). Let D=$\{d_1, \ldots, d_n\}$ be a document collection, where each document d_k is composed of a set sentences S_k=$\{s_{1k}, \ldots, s_{zk}\}$. Documents are composed of a sequence of sentences, each one composed of a set of words. The BOW representation of the j-th sentence s_{jk} belonging to the k-th document d_k of the collection D is the set of all word stems (i.e., terms) occurring in s_{jk}.

Consider now the set tr_{jk}=$\{w_1, \ldots, w_l\}$ where $tr_{jk} \subseteq s_{jk}$ and $w_q \neq w_r$ $\forall\, q \neq r$. It includes the subset of distinct terms occurring in the sentence s_{jk}. To tailor document sentences to the transactional data format, we consider each document sentence as a transaction whose items are distinct terms taken from its BOW representation, i.e., tr_{jk} is the transaction that corresponds to the document sentence s_{jk}. A transactional representation T of the document collection D is the union of all transactions tr_{jk} corresponding to each sentence s_{jk} belonging to any document $d_k \in D$.

The document collection is associated with the statistical measure of the term frequency-inverse document frequency (tf-idf) that evaluates the relevance of a word in the whole collection. A more detailed description of the tf-idf statistic follows. The whole document content could be represented in a matrix form TC, in which each row represents a distinct term of the document collection while each column corresponds to a document. Each element tc_{ik} of the matrix TC is the tf-idf value associated with a term w_i in the document d_k belonging to the whole collection D. It is computed as follows:

$$tc_{ik} = \frac{n_{ik}}{\sum_{r \in \{q \,:\, w_q \in d_k\}} n_{rk}} \cdot \log \frac{|D|}{|\{d_k \in D \,:\, w_i \in d_k\}|} \tag{1}$$

where n_{ik} is the number of occurrences of i-th term w_i in the k-th document d_k, D is the collection of documents, $\sum_{r \in \{q \,:\, w_q \in d_k\}} n_{rk}$ is the sum of the number of occurrences of all terms in the k-th document d_k, and $\log \frac{|D|}{|\{d_k \in D \,:\, w_i \in d_k\}|}$ represents the inverse document frequency of term w_i.

2.2 The pattern-based model generation

Frequent itemset mining is a well-established data mining approach that focuses on discovering recurrences, i.e., itemsets, that frequently occur in the source data. An itemset I of length k, i.e., a k-itemset, is a set of k distinct items. Let T be the document collection in the transactional data format. We denote as $\mathcal{D}(I)$ the set of transactions supported by I, i.e., $\mathcal{D}(I) = \{tr_{jk} \in T \mid I \subseteq tr_{jk}\}$. The support of an itemset I is the observed frequency of occurrence of I in D, i.e., $sup(I)=\frac{\mathcal{D}(\mathcal{I})}{|T|}$. Since the problem of discovering all itemsets in a transactional dataset is computationally intractable [1], itemset mining is commonly driven by a minimum support threshold *min_sup*.

Given a minimum support threshold *min_sup* and a model size p, PatTexSum generates a pattern-based model that includes the most informative yet non-redundant set of p frequent itemsets discovered from the document collection T tailored to the transactional data format (Cf. Section 2.1).

Among the large set of previously proposed approaches focused on succinctly representing transactional data by means of itemsets [8, 17, 18], we adopt a method recently proposed in [11]. Unlike previous approaches, it exploits an entropy-based heuristic to drive the mining process and select most informative yer not redundant itemsets without the need of postpruning. Its efficiency and effectiveness in discovering succinct transactional data summaries makes it particularly suitable for the application to text summarization.

2.3 Sentence evaluation and selection

The PatTexSum method exploits the pattern-based model to evaluate and select most relevant sentences to include in the summary. Sentence evaluation and selection steps consider (i) a sentence relevance score that combines the tf-idf statistic [16] associated with each sentence term, and (ii) the sentence coverage with respect to the generated pattern-based model (Cf. Section 2.2). In the following we formalize both sentence coverage and relevance.

Sentence relevance score

The relevance score of a sentence is evaluated by using the bag-of-word document representation. It is computed as the sum of the tf-idf values (Cf. Formula 1) of each term belonging to the sentence in the document collection. In Formula 2 the score expression for a generic sentence s_{jk} belonging to the document collection D is reported

$$SR(s_{jk}) = \frac{\sum_{i \ | \ w_i \in s_{jk}} tc_{ik}}{|t_{jk}|} \tag{2}$$

where $|t_{jk}|$ is the number of distinct terms occurring in s_{jk}, and $\sum_{i \ | \ w_i \in s_{jk}} tc_{ik}$ is the sum of the tf-idf values associated with terms (i.e., word stems) in s_{jk} (Cf. Formula 1).

Sentence model coverage

The sentence coverage measures the pertinence of each sentence to the generated pattern-based model. To this aim, it considers document sentences tailored to the transactional data format. Let D be the collection of documents, i.e., a set of sentences. We first associate with each sentence $s_{jk} \in D$ a binary vector, denoted in the following as *sentence coverage vector* (SC), SC_{jk}=$\{sc_1, \ldots, sc_p\}$ where p is the number of itemsets belonging to the model and $sc_i = \mathbf{1}_{tr_{jk}}(I_i)$ indicates whether itemset I_i is included or not in tr_{jk}. More formally, $\mathbf{1}_{tr_{jk}}$ is an indicator function defined as follows:

$$\mathbf{1}_{tr_{jk}}(I_i) = \begin{cases} 1 & \text{if } I_i \subseteq tr_{jk}, \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The coverage of a sentence s_{jk} with respect to the pattern-based model is defined as the number of 1's that occur in the corresponding coverage vector SC_{jk}.

We formalize the problem of selecting the most informative and not redundant sentences according to the pattern-based model as a set covering problem.

The set covering problem

A set covering algorithm focuses on selecting the minimum set of sentences, of arbitrary size l, whose logic OR of coverage vectors, i.e., SC^{*}=$SC_1 \vee \ldots \vee SC_l$, generates a binary vector composed of all 1's. This implies that each itemset belonging to the model covers at least one sentence. The SC^{*} vector will be denoted as the *summary coverage vector* throughout the paper.

Algorithm 1 Sentence selection - Greedy approach

```
Input: set of sentence relevance scores $SR$, set of sentence coverage vectors $SC$, tf-idf matrix $TC$
Output: summary $\mathcal{S}$
 1: {Initializations}
 2: $\mathcal{S} = \emptyset$
 3: $ESC = \emptyset$ {set of eligible sentence coverage vectors}
 4: $SC^* =$ all_zeros() {summary coverage vector with only 0s}
 5: {Cycle until either $SC^*$ contains only 1s or all the $SC$ vectors contain only zeros}
 6: while not (summary_coverage_vector_all_ones() or sentence_coverage_vectors_only_zeros()) do
 7:   {Determine the sentences with the highest number of ones}
 8:   $ESC =$ max_ones_sentences()
 9:   if $ESC$ != $\emptyset$ then
10:     {Select the sentence with maximum relevance score}
11:     $SC_{best} = ESC[1]$
12:     for all $t \in ESC[2:]$ do
13:       if $SR_t > SR_{best}$ then
14:         $SC_{best}{=}SC_t$
15:       end if
16:     end for
17:     {Update sets and summary_coverage_vector}
18:     $\mathcal{S} = \mathcal{S} \cup SC_{best}$
19:     $SC^*{=}\ SC^*$ OR $SC_{best}$
20:     $ESC = ESC \setminus SC_{best}$
21:     {Update the sentence coverage vectors belonging to $\mathcal{V}$}
22:     for all $SC_i$ in $SC$ do
23:       $SC_i = SC_i$ AND $\overline{SC^*}$
24:     end for
25:   else
26:     break
27:   end if
28: end while
29: return $\mathcal{S}$
```

The set covering problem is known to be NP-hard. To solve the problem, we adopt a greedy strategy that we already proved to be effective in summarization of biological microarray data [2]. In order to build an accurate yet concise summary, the sentence coverage with respect to the pattern-based model is considered as the most discriminative feature, i.e., sentences that cover the maximum number of itemsets belonging to the model are selected firstly. At equal terms, the sentence with maximal coverage that is characterized by the highest relevance score SR is preferred.

The adopted 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^*. The pseudo-code of the greedy approach is reported in Algorithm 1. It takes in input the set of sentence relevance scores SR, the set of sentence coverage vectors SC, and the tf-idf matrix TC. It produces the summary $\mathcal{S}$, i.e., the minimal subset of most representative sentences. The first step is the variable initialization and the sentence coverage vector computation (lines 1-4). Next, the sentence with maximum coverage, i.e., the one whose coverage vector contains the highest number of ones, is iteratively selected (line 7). At equal terms, the sentence with maximum relevance score (Cf. Formula 2) is preferred (lines 12-16). Finally the selected sentence is included in the summary $\mathcal{S}$ while the summary and sentence coverage vectors are updated (lines 18-24). The procedure iterates until either the summary coverage vector contains only ones, i.e., the model is fully covered by the summary, or the remaining sentences are not covered by any itemset, i.e., the remaining sentences are not pertinent to the model (line 6).

Experimental results, reported in Section 3, show that the proposed summarization method performs better than exclusively considering either sentence coverage or sentence relevance.

3 Experimental results

We conducted a set of experiments to address the following issues: (i) the effectiveness of the proposed summarization approach against two widely used summarizers, i.e., the Open Text Summarizer (OTS) [14] and TexLexAn [19] (Section 3.1), and (ii) the impact of the pattern-based model size and the support threshold on the performance of PatTexSum (Section 3.2).

We evaluated all the summarization approaches on a collection of real-life news articles. To this aim, the 10 top-ranked news documents, provided by the Google web search engine (http://www.google.com), that concern the following recent news topics have been selected:

- **Natural Disaster**: Earthquake in Spain 2011
- **Royal Wedding**: Prince William and Kate Middleton wedding
- **Technology**: Microsoft purchased Skype
- **Education**: Wealthy parents could buy their children places at elite universities
- **Sport**:Australia defeat Pakistan in Azlan shah Hockey

The datasets relative to the above news categories are made available for research purposes, upon request to the authors.

To compare the results by PatTexSum with OTS [14] and TexLexAn [19], we used the ROUGE [10] toolkit (version 1.5.5), which is widely applied by Document Understanding Conference (DUC) for document summarization performance evaluation[1]. It measures the quality of a summary by counting the unit overlaps between the candidate summary and a set of reference summaries. Intuitively, the summarizer that achieves the highest ROUGE scores could be considered as the most effective one. Several automatic evaluation scores are implemented in ROUGE. For the sake of brevity, we reported only ROUGE-2 and ROUGE-4 as representative scores. Analogous results have been obtained for the other scores.

Since a ”golden summary” (i.e., the optimal document collection summary) is not available for web news document, we performed a leave-one-out

[1] The provided command is: 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

dataset	PatTexSum				**OTS**			**TexLexAn**		
	p	R	Pr	F	R	Pr	F	R	Pr	F
Natural Disaster	16	**0.116**	**0.288**	**0.141**	0.040	0.120	0.053	0.038	0.114	0.045
Royal Wedding	12	**0.036**	**0.215**	**0.058**	0.034	0.174	0.054	0.030	0.150	0.047
Technology	5	**0.141**	**0.465**	**0.210**	0.042	0.208	0.067	0.042	0.172	0.065
Sports	10	**0.145**	**0.297**	**0.189**	0.055	0.133	0.075	0.071	0.149	0.093
Education	8	**0.039**	**0.241**	**0.064**	0.036	0.170	0.054	0.034	0.150	0.051

Table 1 Performance comparison in terms of ROUGE-2 score.

cross validation. More specifically, for each category we summarized nine out of ten news documents and we compared the resulting summary with the remaining (not yet considered) document, which has been selected as golden summary at this stage. Next, we tested all other possible combinations by varying the golden summary and we computed the average performance results, in terms of precision, recall, and F-measure, achieved by each summarizer for both ROUGE-2 and ROUGE-4.

3.1 Performance comparison and validation

We evaluated the performance, in terms of ROUGE-2 and ROUGE-4 precision (Pr), recall (R), and F-measure (F), of PatTexSum against OTS and TexLexAn. For both OTS and TexLexAn we adopted the configuration suggested by the respective authors. For PatTexSum we enforced a minimum support threshold *min_sup*=1.5% and we tuned the value of the pattern-based model size p to its best value for each considered dataset. A more detailed discussion on the impact of both *min_sup* and p on the performance of PatTexSum is reported in Section 3.2.

PatTexSum performs better than the other considered summarizers on all tested datasets. To validate the statistical significance of PatTexSum performance improvement against OTS and TexLexAn, we used the paired t-test [7] at significance level $p - value = 0.05$ for all evaluated datasets and measures. For ROUGE-2, PatTexSum provides significantly better results than OTS, whose summarization approach is mainly based on tf-idf measure, and TexLexAn in terms of precision and/or recall on 3 out of 5 datasets (i.e., Natural disaster, Technology and Sports). Moreover, PatTexSum significantly outperforms TexLexAx and OTS in terms of F-measure (i.e., the harmonic average of precision and recall [16]) on, respectively, 2 and 3 of them (i.e., Natural disaster and Technology for both, and Sports for TexLexAn). Similar results were obtained for ROUGE-4.

dataset	PatTexSum				**OTS**			**TexLexAn**		
	p	R	Pr	F	R	Pr	F	R	Pr	F
Natural-Disaster	16	**0.060**	**0.125**	**0.068**	0.005	0.012	0.006	0.005	0.011	0.006
Royal-wedding	12	**0.009**	**0.082**	**0.015**	0.003	0.018	0.005	0.003	0.018	0.005
Technology	5	**0.113**	**0.356**	**0.167**	0.009	0.065	0.016	0.003	0.011	0.005
Sports	10	**0.059**	**0.112**	**0.077**	0.004	0.010	0.006	0.022	0.036	0.027
Education	8	**0.017**	**0.141**	**0.030**	0.003	0.012	0.005	0.003	0.009	0.004

Table 2 Performance comparison in terms of ROUGE-4 score.

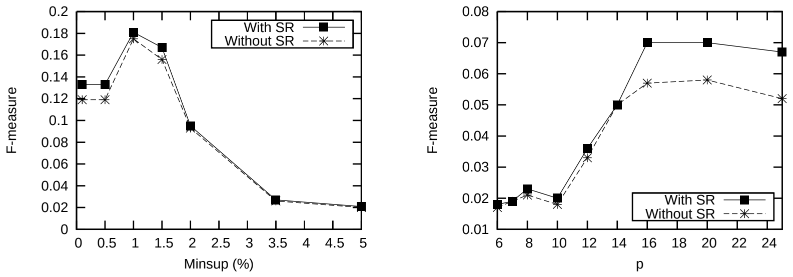
3.2 PatTexSum ***parameter analysis***

We analyzed the impact of the minimum support threshold and the pattern-based model size, i.e., the number of generated itemsets, on the performance of the PatTexSum summarizer. To also test the impact of the tf-idf statistic on the performance of the pattern-based summarizer, we entail (i) neglecting the relevance score evaluation (i.e., by simply selecting the top-ranked maximal coverage sentence provided by the itemset miner [11]), and (ii) considering other statistical measures in place of the tf-idf score. Among all the evaluated scores, the tf-idf statistic turns out to be most effective measure in discriminating among sentences.

In Figures 2(a) and 2(b) we reported the F-measure achieved by PatTexSum, by either considering or not the relevance score in the sentence evaluation, and by varying, respectively, the support threshold on Technology and the model size on the Natural Disaster document collection. For the sake of brevity, we reported only the results obtained with the ROUGE-4 score. Analogous results have been obtained for the other ROUGE scores, for precision and recall measures, and for all other configurations.

The usage of the relevance score based on the tf-idf statistic always improves the performance of PatTexSum in the range of those values of p and min_sup yielding the highest F-measure. This improvement is due to its ability to well discriminate sentence term occurrence among documents. When higher support thresholds (e.g., 5%) are enforced, many informative patterns are discarded, thus the model becomes too general to yield high summarization performance. Oppositely, when very low support thresholds (e.g., 0.1%) are enforced, data overfitting occurs, i.e., the model is too much specialized to effectively and concisely summarize the whole document collection content. At medium support thresholds (e.g., 1.5%) the best balancing between model specialization and generalization is achieved, thus, PatTexSum produces very concise yet informative summaries.

The model size may also significantly affect the summarization performance. When a limited number of itemsets (e.g., $p = 6$) is selected, the relevant knowledge hidden in the news category Natural Disaster is not yet fully covered by the extracted patterns (see Figure 2(b)), thus the generated summaries are not highly informative. When $p = 16$ the pattern-based



(a) Technology. p=5. Impact of the support threshold.

(b) Natural Disaster. min_sup=1.5%. Impact of the pattern-based model size.

Fig. 2 PatTexSum performance analysis by either considering or not of the relevance score (SR). Rouge-4 score. F-measure.

model provides the most informative and non-redundant knowledge. Consequently, the multi-document pattern-based summarization becomes very effective. When a higher number of itemsets is included in the model, the quality of the generated summaries worsens as the model is still informative but redundant. The best values of model size and support threshold achieved by each news category depend on the analyzed document term distribution.

4 Conclusions and future works

This paper presents a multi-document summarizer that combines the knowledge provided by a pattern-based model, composed of frequent itemsets, with a statistical evaluation, based on the well-founded tf-idf measure, to select the most representative and not redundant sentences. Albeit the application of frequent itemsets to represent most valuable correlations among transactional data is well-established, their usage in text summarization has never been investigated so far. The proposed summarizer exploits a greedy approach to combine knowledge discovered from two different data representations, i.e., the transactional and bag-of-word representations, and select the minimal set of most relevant sentences. Experiments conducted on real-life news articles show both the effectiveness and the efficiency of the proposed text summarization method.

Future works will address: (i) the extension of the proposed approach to address the problem of incremental summary updating, and (ii) the exploitation of new techniques to address the set covering problem.

References

1. R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. In *ACM SIGMOD Record*, volume 22, pages 207–216.
2. E. Baralis, G. Bruno, and A. Fiori. Minimum number of genes for microarray feature selection. *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC-08)*, pages 5692–5695, 2008.
3. S. Bird, E. Klein, and E. Loper. *Natural language processing with Python*. O'Reilly Media, 2009.
4. S. Brin, R. Motwani, and C. Silverstein. Beyond market baskets: Generalizing association rules to correlations. In *SIGMOD Conference*, pages 265–276, 1997.
5. S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the seventh international conference on World Wide Web 7*, pages 107–117, 1998.
6. J. M. Conroy, J. Goldstein, J. D. Schlesinger, and D. P. Oleary. Left-brain/right-brain multi-document summarization. In *In Proceedings of the Document Understanding Conference*, 2004.
7. T. G. Dietterich. Approximate statistical test for comparing supervised classification learning algorithms. *Neural Computation*, 10(7), 1998.
8. S. Jaroszewicz and D. A. Simovici. Interestingness of frequent itemsets using bayesian networks as background knowledge. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 178–186, 2004.
9. K.-N. Kontonasios and T. D. Bie. An information-theoretic approach to finding informative noisy tiles in binary databases. In *SIAM International Conference on Data Mining*, pages 153–164, 2010.
10. C.-Y. Lin and E. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 71–78, 2003.
11. M. Mampaey, N. Tatti, and J. Vreeken. 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*, 2011.
12. D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:2004, 2004.
13. D. R. Radev, H. Jing, M. Stys, and D. Tam. Centroid-based summarization of multiple documents. *Information Processing and Management*, 40(6):919 – 938, 2004.
14. N. Rotem. Open text summarizer (ots). *Retrieved July*, 3(2006):2006, 2003.
15. H. Takamura and M. Okumura. Text summarization model based on the budgeted median problem. In *Proceeding of the 18th ACM conference on Information and knowledge management*, pages 1589–1592, 2009.
16. P. Tan, M. Steinbach, V. Kumar, et al. *Introduction to data mining*. Pearson Addison Wesley Boston, 2006.
17. N. Tatti. Probably the best itemsets. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 293–302, 2010.
18. N. Tatti and H. Heikinheimo. Decomposable families of itemsets. In *Proceedings of the European conference on Machine Learning and Knowledge Discovery in Databases - Part II*, pages 472–487, 2008.
19. TexLexAn. Texlexan: An open-source text summarizer, 2011.
20. D. Wang and T. Li. Document update summarization using incremental hierarchical clustering. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 279–288, 2010.