Utilizing User Posts to Enrich Web Document Summarization with Matrix Co-factorization

Minh-Tien Nguyen*
Japan Advanced Institute of Science and Technology.
1-8 Asahidai, Nomi, Ishikawa, Japan.
tiennm@jaist.ac.jp

Tran Viet Cuong
Hanoi University of Science and Technology.
Hanoi, Vietnam.
cuongti1100@gmail.com

Nguyen Xuan Hoai
IT R&D Center, Hanoi University and
R&D Department, Anvita Jsc.
Hanoi, Vietnam.
nxhoai@hanu.edu.vn

Minh-Le Nguyen
Japan Advanced Institute of Science and Technology.
1-8 Asahidai, Nomi, Ishikawa, Japan.
nguyenml@jaist.ac.jp

ABSTRACT

In the context of social media, users tend to post relevant information corresponding to an event mentioned in a Web document. This paper presents a model to capture the nature of the relationships between sentences and user posts such as relevant comments in sharing hidden topics for enriching summarization. Unlike the previous methods which usually base on hand-crafted features, our approach ranks sentences and comments based on their importance affecting the topics. The sentence-comment relation is formulated in a share topic matrix, which presents their mutual reinforcement support. Our newly proposed matrix co-factorization algorithm computes the score of each sentence and comment and extracts top m ranked sentences and m comments as the summarization. Experimental results on two datasets in English and Vietnamese of the social context summarization task and DUC 2004 confirm the efficiency of our model in summarizing Web documents.

CCS CONCEPTS

• **Information systems** → **Summarization**; • **Computing methodologies** → *Natural language processing*;

KEYWORDS

Web Document Summarization, Matrix Factorization.

ACM Reference Format:
Minh-Tien Nguyen, Tran Viet Cuong, Nguyen Xuan Hoai, and Minh-Le Nguyen. 2017. Utilizing User Posts to Enrich Web Document Summarization with Matrix Co-factorization. In *SoICT '17: Eighth International Symposium on Information and Communication Technology, December 7–8, 2017, Nha Trang City, Viet Nam.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3155133.3155196

*The Center of Scientific Research and Technological Applications, Hung Yen University of Technology and Education, Vietnam.

SoICT '17, December 7–8, 2017, Nha Trang City, Viet Nam

ACM ISBN 978-1-4503-5328-1/17/12…$15.00
https://doi.org/10.1145/3155133.3155196

1 INTRODUCTION

Text summarization is a process to reduce the size of an input document while preserving its important content [3, 7]. The reduction falls into extraction or abstraction. The extraction tries to extract a small number of sentences in a document. On the other hand, the abstraction generates the summarization which does not completely use the words or phrases in the original document. The example of abstraction is that writers usually create highlights (short sentences which capture major information) for each Web document. While the extraction achieves promising results, the output of the abstraction is still far from the human satisfaction. The summarization in fact benefits many natural language processing applications such as Web search or highlight generation.

In the context of social media, users have an ability to freely discuss events and topics mentioned by a news provider by posting their messages. For instance, Yahoo News provides both news articles along with a Web interface where readers can write their comments about an event such as `"Germanwings crash families could seek damages in the U.S.: lawyer"`. These comments seen as a form of user posts, also called by social information [28, 38, 42], not only reflect the news article contents but also convey readers' viewpoints. An example in Table 1[1] indicates that (i) the sentence is important because it includes important information for inferring the event. (ii) The comment also contains salient words or phrases, which can be used to enrich the information in the sentence. Furthermore, with a formal writing style, some comments can be directly used as the summarization as well as sentences. This raises a challenging task of how to exploit relevant user posts to enrich the summarization of the main document.

Table 1: An extraction example.

[S]: **Families** of the **victims** of the **Germanwings** crash are considering filing a **claim for damages** in the United States if they cannot reach agreement with parent airline Lufthansa in Germany, a lawyer representing the families said on Sunday.
[C]: I'm sad for the people who are no longer on their life of this **Germanwings** strategy, but I don't know how much **money relatives** want for the flesh of **who die** on **Germanwings** case.

[1]https://www.yahoo.com/news/germanwings-crash-families-could-seek-damages-u-lawyer-140040694–sector.html

In this paper, we propose a novel summarization approach based on non-negative matrix co-factorization to automatically extract summary sentences and comments of a Web document by incorporating its social context. The motivation for our model comes from the fact that sentences and comments share hidden topics in term of common words or phrases. For example, in Table 1, an extracted sentence and comment share common or inferred words in bold, e.g. *"victims"* ∼ *"who die"*, *"Germanwings"*, *"families"* ∼ *"relatives"*. These terms form hidden topics presenting the nature of relationships between sentences and comments. With this observation, we represent sentences and comments in a share topic-matrix, which formulates the common topics in a mutual information fashion. The summarization is extracted by analyzing the matrix using a non-negative matrix co-factorization approach. The main contributions of the paper are as follows:

- Our new approach simulates the natural mutual relationship between sentences and comments in sharing hidden topics represented by a share topic-matrix. The matrix is found and used by matrix co-factorization to find salient sentences and comments. To the best of our knowledge, we are the first to use matrix co-factorization for the task.
- It investigates the normalization to avoid over-fitting during the optimization.
- It carries out extensive experiments on two datasets in social context summarization and one benchmark dataset. ROUGE-scores validate the efficiency of our model.
- It presents a robust model for summarizing Web documents, which include user posts. Our model is unsupervised suggesting that it can be applied to unrestricted domains. Our code is publicly accessible.[2]

We apply our model to the task of sentence extraction on two datasets in two languages (English and Vietnamese), and also on DUC 2004. ROUGE-scores indicate that: (i) our model obtains sufficient improvements over strong methods in social context summarization and (ii) it is competitive on DUC 2004.

2 RELATED WORK

Extractive summarization has been extensively studied in the literature [4, 7, 14, 21, 23, 25, 31, 35, 40, 41]. It usually formulates the extraction as a ranking [7, 23], defining concepts and solving an optimization [40, 41], supervised learning in form of binary classification [14, 31, 35], or deep learning [4, 25]. Recently, matrix factorization has also been applied to the extraction [10, 17, 33, 37]. Wang et al. presented a multi-document summarization model which analyzes sentence-level semantic matrix by using symmetric non-negative matrix factorization (NMF) [37]. The summarization is formed by selecting the most informative sentences in each cluster. In this work, we extend the NMF for the task by exploiting relevant user posts for Web document summarization. Our model is different from [37] in that it integrates user posts into the summary process and presents a matrix co-factorization algorithm for extracting the summarization.

Social information has been widely investigated for Web document summarization [1, 13, 22, 36, 42]. For instance, Yang et al. introduced a dual wing factor graph model, which presents the link between sentences and tweets [42]. The authors use SVM and CRF as preliminary steps for building the graph. The summarization is extracted by using an optimization. Wei and Gao used a learning to rank approach with local and cross features trained by RankBoost for news highlight extraction [38]. By contrast, several unsupervised models have been presented [9, 18, 39]. For example, Gao et al. proposed a cross-collection topic-aspect as a preliminary step for a co-ranking method to extract sentences and tweets [9]. Wei and Gao adapted LexRank to build a heterogeneous graph random walk to summarize single documents [39].

SoRTESum system [28, 29] is perhaps the most relevant work to our study. It attempts to model sentence-tweet relationship by using a set of lexical-level features to score sentences and tweets. The score of a sentence or tweet is computed in a reinforcement fashion. Finally, SoRTESum extracts top m sentences and tweets, which have the highest scores. However, these sentence-tweet features are usually hand-crafted, domain-dependent, and difficult to represent the nature of relationships between sentences and comments. Our model distinguishes [28, 29] in which it exploits the share topics between sentences and comments to rank them by using non-negative matrix co-factorization instead of using feature-driven ranking, which is limited with domain adaptation.

3 PRELIMINARIES

3.1 Social context

There are several ways to define the context of a Web document such as using hyperlinks [1], click-through data [36], or user-generated content [13, 28, 29, 38, 42]. In this paper, we follow Yang et al. [42] to define the context of a Web document. Given a Web document d, its social context is C_d represented by $\langle S_d, UP_d, U_d \rangle$, where S_d is a set of sentences in the document d, UP_d is a set of user posts, e.g. comments written by users U_d on d. In this work, we eliminate the user aspect because it is an implicit factor and is unavailable in datasets. We leave the usage of U_d as a feature task.

3.2 The task

Given a document d and its context C_d, the summarization is to extract m important sentences and m representative social messages [28, 38, 42], see Table 1. The intuition of this extraction is that while extracted sentences include salient information, user posts such as comments can also be used to enrich sentences. The enrichment provides new information which is not usually available in extracted sentences as well as the main document.

4 SUMMARIZATION WITH MATRIX CO-FACTORIZATION

This section describes our propsosed method, which is presented in two parts - data preparation and summarization with matrix co-factorization.

4.1 Data Preparation

Since DUC datasets[3] lack user posts (social information); therefore, we have prepared two other ones in two languages for the task:

[2] https://drive.google.com/file/d/0ByufX58sGMBIaElfcGREOFVMZU0/view?usp=sharing

[3] http://duc.nist.gov/data.html

SoLSCSum. is an English dataset created for social context summarization in [30]. The dataset contains 157 open-domain news articles along with 3,462 sentences, 5,858 gold-standard references, and 25,633 comments collected from Yahoo News.[4] The label was created based on majority voting among annotators for training supervised learning methods.

VSoLSCSum. is a labeled Vietnamese dataset created for social context summarization [27]. The dataset consists of 141 open-domain articles along with 3,760 sentences, 2,448 gold-standard references, and 6,926 comments in 12 events. The articles are collected from several Vietnamese web pages such as DanTri.[5]

Table 2: Statistical observation on the two datasets.

Dataset	#doc	#sentences	#comments	#references
SoLSCSum	157	3,462	25,633	5,858
VSoLSCSum	141	3,760	6,926	2,448

Table 2 shows observation on the two datasets. The number of comments is considerable to provide additional information for supporting sentences. The number of gold-standard references in both datasets is quite large because it contains both selected sentences and comments. The annotators select important sentences and comments (labeled by 1) to form the references due to the aim of extractive summarization on the two datasets.

4.2 Summarization

This sections first introduces document representation. It next shows the basic and novel model for the summarization.

4.2.1 Document representation. Given a document d with t terms and n sentences, we can use a term-sentence matrix X to represent d [10]. In this model, $X(i, j)$ is the weight of term t_i in sentence s_j computed by term frequency (TF). However, for our purpose, this representation is defficient since $X(i, j)$ is only calculated on sentence level, which ignores the document level. We, therefore, adopt the approach in [24], in which it considers both local and global weights of a term.

$$X(i, j) = L(i, j) \times G(i) \tag{1}$$

where $L(i, j)$ is the local weight (TF) and $G(i)$ is the global weight (document inverse frequency - IDF). In other words, Eq. (1) can be referred as TF-IDF. This representation is employed in both basic and advanced models given below.

4.2.2 Basic model. We start with a basic model using non-negative matrix factorization (NMF) [15] to summarize documents [10, 17, 33, 37]. Given a term-sentence matrix X, NMF finds two non-negative matrices W and H [19] such that:

$$X \approx WH \tag{2}$$

where W is the topic matrix, H is the sentence weight matrix presenting the effect/importance of sentences to the topics given in W. To find W and H, an optimization procedure is needed for minimizing the following error function:

$$E = |X - WH|_F^2 \tag{3}$$

If each column of X represents a sentence (an object), NMF approximates its linear decomposition in the bases of k topics (where k is the column size of W). The weight score, WS_j, of each sentence is calculated as follows:

$$WS_j = \sum_{i=1}^{k} H_{ij} \times weight(H_i) \tag{4}$$

$$weight(H_i) = \frac{\sum_{q=1}^{n} H_{iq}}{\sum_{p=1}^{k} \sum_{q=1}^{n} H_{pq}} \tag{5}$$

where k is the chosen topic number and n is the number of sentences. After having the weight score of each sentence, WS_j, we can rank the sentences based on their scores on the matrix H and extract top m sentences for a summary.

4.2.3 Novel model with co-factorization. The basic model only uses sentences to create the term-sentence matrix. It ignores comments, which can be used to enrich information of the main text. Here, we assume that the comments and the main text share hidden topics in form of common words or phrases (42.05% words in comments are from the main text for SoLSCSum and 44.82% for VSoLSCSum). Consequently, we propose a co-factorization for the representation of the main text and comments.

Matrix creation. Given a document d with n main text sentences and m comment sentences, we use two term-sentence matrices: X_1 and X_2 for representating the main text and comments respectively. To create these matrices, we first merged the sentences and comments into a single set and generated a term dictionary from this set by using word lemmatization, stemming, and stopword removal from `NLTK`,[6].[7] Suppose the size of the dictionary is u, hence $X_1 = u \times n$ and $X_2 = u \times m$ ($n \ll u$ and $m \ll u$). We applied Eq. (1) to each component of X_1 and X_2 to compute their weights. The TF of a term w is computed on the sentence level, whereas its IDF is calculated on the document level. For example, if w appears in sentences, we consider each sentence as a single document and count the number of sentences including w as document frequency (DF). If w appears in both document d and its comments, two DF values are separately calculated for each side. This creation exploits comments for creating the term-sentence matrix, which encodes the share hidden topics into the summarization.

Non-negative matrix co-factorization (NMCF). Given X_1 and X_2, we can indepently apply NMF to X_1 and X_2. However, as in the aforementioned assumption that they share the hidden topics, they can be jointly optimized in a unified co-factorization framework. Let k be the number of the share hidden topics between the main text sentences and comments, we can rewrite Eq. (2) to represent the co-factorization of X_1 and X_2 as follows:

$$X_1 \approx WH_1 \tag{6}$$

$$X_2 \approx WH_2 \tag{7}$$

where $W = u \times k$ is the matrix of the share hidden topics; $H_1 = k \times n$ is the sentence weight matrix, which represents the effect of the main text sentences to the hidden topics W; and $H_2 = k \times m$ is the comment weight matrix, which represents the effect of comments

[4] https://www.yahoo.com/news/
[5] http://dantri.com.vn
[6] http://www.nltk.org
[7] We did not do lemmatization and stemming on VSoLSCSum since there are no effetive algorithms for Vietnamese at the moment.

to W. The sensitivity analysis of selecting an appropriate topic number k is shown in §5.3.

For the co-factorization, the error function in Eq. (3) can be redefined as follows:

$$E = |X_1 - WH_1|_F^2 + |X_2 - WH_2|_F^2 \tag{8}$$

The new error function includes two components: one for main text sentences and the other for comments. The optimization procedure has to consider both components to achieve the global optimum instead of optimizing only one component as in Eq. (3). Eq. (8) also reveals an important characteristic of our model, in which sentences and comments are formulated in a mutual reinforcement support. To minimize Eq. (8), one can utilize several techniques such as multiplicative update method [16], alternating non-negative least squares [16, 19], and gradient approaches [16, 19]. We choose to adapt the gradient algorithm [16, 19] for our co-factorization approach as it has been shown to be efficient in the literature. Eqs. (9)–(11) describe the gradient calculation in our proposed optimization algorithm.

$$\frac{\nabla E}{\nabla W} = (WH_1 - X_1)H_1^T + (WH_2 - X_1)H_2^T \tag{9}$$

$$\frac{\nabla E}{\nabla H_1} = W^T(WH_1 - X_1) \tag{10}$$

$$\frac{\nabla E}{\nabla H_2} = W^T(WH_2 - X_2) \tag{11}$$

with the update rules as follows:

$$W = W \odot \frac{X_1H_1^T + X_2H_2^T}{WH_1H_1^T + WH_2H_2^T} \tag{12}$$

$$H_1 = H_1 \odot \frac{W^TX_1}{W^TWH_1} \tag{13}$$

$$H_2 = H_2 \odot \frac{W^TX_2}{W^TWH_2} \tag{14}$$

where $\odot$ is the Hadamard product. To ensure the uniqueness of NMF, the normalization of W and H (H_1 and H_2) could be utilized with L_1 or L_2.

Normalization L_1:

$$w_{ij} = \frac{w_{ij}}{\sum_i w_{ij}} \tag{15}$$

$$h_{ij} = h_{ij} \sum_i w_{ij} \tag{16}$$

or L_2:

$$w_{ij} = \frac{w_{ij}}{\sqrt{\sum_i w_{ij}^2}} \tag{17}$$

$$h_{ij} = h_{ij} \sqrt{\sum_i w_{ij}^2} \tag{18}$$

The effects of using L_1 or L_2 are analyzed in §5.4.

Sentence selection. After the optimization procedure finds the optimal solutions, we apply Eqs. (4) and (5) to W and H_1 to select m important sentences (having the highest weights); and to W and H_2 to extract m representative comments.

5 RESULTS AND DISCUSSION

5.1 Settings and evaluation metric

Setting and parameters. We used 10-fold cross-validation for SoLSCSum with $m = 6$ as in [30]; 5-fold cross-validation for VSoLSCSum with $m = 6$ for VSoLSCSum as in [27].[8] Stop words and links were removed, except for VSoLSCSum due to there is no known formal stop word list in Vietnamese.

Baselines. We first compare our new co-factorization approach to the basic **NMF** with the same parameter setting (e.g k).

We also compare our **NMCF** to the models in social context summarization. They include three basic models: (i) **Lead-*m***: chooses the first m sentences[26] as the summarization; (ii) **LexRank**[9] [8] uses a stochastic graph-based method for computing relative importance of textual units in text summarization; and (iii) **SoviVote** uses Cosine voting from comments to score sentences and vice versa [39]. We also compare **NMCF** to three advanced models, including: (iv) **cc-TAM** uses a cross-collection topic-aspect modeling (cc-TAM) as a preliminary step to generate a bipartite graph used for co-ranking [9]; (v) **HGRW** is a variation of LexRank named Heterogeneous Graph Random Walk [39]; and (vi) **SoRTESum** builds a similarity graph with a set of similarity features [28]. The score of a sentence is computed by only using social context of a document (SoRTESum Inter Wing) from comments, or the combination of internal and social information (Dual Wing). After modeling, the baselines were separately applied to documents and their comments to extract m salient sentences and m comments as the summarization.

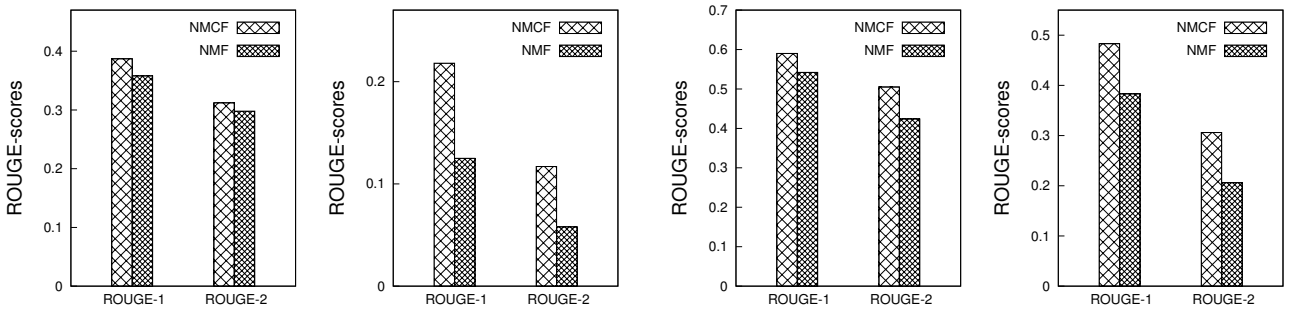
Evaluation metric. Extracted sentences and comments were compared to gold-standard references by using ROUGE-N (N=1, 2 with F-score) [20]. ROUGE-N computes n–grams word overlapping between extracted summaries and gold-standard references. We employed `ROUGE1.5.5` with the wrapper of `python` in `pyrouge`[10] package.[11] Due to different length of summaries and references we use F-score to balance between precision and recall. We separately evaluated the extracted sentences and comments instead of combining them as the setting in [9, 28, 38, 42].

5.2 Experimental Results

We first present the comparison of our NMCF against NMF. We next show the ROUGE-scores of our model and advanced methods on the two datasets as well as on DUC 2004.

NMCF vs. NMF. Figure 1 presents the ROUGE-scores of our model and NMF. It is clear from the figure that our NMCF method obtains significant performance improvements over NMF in almost all cases (p-values ≤ 0.05). For example, for sentence extraction on VLSCSum in Figure 1d, the performance of our method is 5% higher than NMF in ROUGE-1 and 12% in ROUGE-2. This trend is consistent in comment extraction on the two datasets in Figures 1e and 1b, in which there are big margins between our method and NMF (around 10%). This confirms the efficiency of our method in exploiting mutual information between comments and sentences

[8]We gratefully acknowledge [30] and [27] for sharing data partition.
[9]https://code.google.com/p/louie-nlp/source/browse/trunk/louie-ml/src/main/java/org/louie/ml/lexrank/?r=10
[10]https://pypi.python.org/pypi/pyrouge/0.1.0
[11]Parameters: -c 95 -2 -1 -U -r 1000 -n 2 -w 1.2 -a -s -f B



(a) Sentence selection (b) Comment extraction (c) ROUGE-scores on SoLSCSum.

(d) Sentence selection (e) Comment extraction (f) ROUGE-scores on VSoLSCSum.

Figure 1: The ROUGE-scores of NMCF and NMF on two datasets.

in the matrix co-factorization. In Figure 1a, NMCF slightly outperforms NMF, e.g. 0.387 vs. 0.358 in ROUGE-1. This is because NMF also uses the advantage of matrix factorization, which has been shown to be efficient for document summarization [10, 17, 33, 37]. In addition, sentences themselves contain important information for summarization; hence, adding more information from comments only slightly improves ROUGE-scores. However, for comment extraction, the support from sentences boosts the ROUGE-scores.

We conducted statistical tests (pairwise $t-$test)[12] to confirm the significance of these improvements. The p-values in Table 3 statistically confirm the efficiency of our model, in which it significantly outperforms NMF in almost all cases. For sentence extraction on SoLSCSum, NMF is competitive with our model in ROUGE-1 and ROUGE-2 with $p-$value = 0.2868 and 0.3277.

Table 3: Two-tailed t-test of NMCF and NMF on two datasets. Bold is significant with $p \leq 0.05$.

Data	Sentence		Comment	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
SoSLCSum	0.2868	0.3277	**0.0000**	**0.0000**
VSoLSCSum	**0.0384**	**0.0008**	**0.0006**	**0.0000**

NMCF vs. baselines and advanced methods. We show the ROUGE-scores of our NMCF and other methods on the two datasets. To be fair, we first compare ours with unsupervised learning methods. ROUGE-scores from Table 5 indicate that our NMCF outperforms the strong methods in almost all cases with some of them are significant (denoted by †). For example, it obtains the best ROUGE-1 of sentence selection on SoLSCSum. There is a small margin between our method and the second best, e.g. 0.387 vs. 0.379. This trend is consistent in the other cases. This is because our method exploits the support of comments and the advantage of matrix co-factorization. In the first aspect, comments help to enrich information of sentences in document representation. A bigger dictionary created from sentences and comments enriches the matrix representation, which improves the scoring. In the second aspect, scoring with matrix co-factorization exploits the share topic-matrix to effectively rank sentences and comments. Our method is the second best in two cases: ROUGE-2 of sentence selection on SoLSCSum and ROUGE-2 of comment extraction on VSoLSCSum. In the first case, Lead-m is a strong baseline, which outperforms many methods participated in DUC competition [26]. In the second case, HGRW is one of the state-of-the-art unsupervised models. It exploits comments to score sentences and uses random walk algorithms to rank sentences [39]. This explains that HGRW is the second best in almost cases. However, in other cases, our NMCF surpasses all other methods validating our idea stated in §1.

We also challenge our NMCF by comparing it with supervised learning methods. The methods include: (i) a learning to rank (L2R) method trained by RankBoost implemented in RankLib[13] with local features cross features (CCF) [38]; (ii) SVM uses local features [42] with RBF kernel; (iii) Ranking SVM[14] uses many refined local and social features [30]. We only report ROUGE-2 due to its ability to compare summarization systems [2, 20, 32] and space limitation.

Table 4: ROUGE-2 of NMCF vs. supervised methods.

Method	SoLSCSum		VSoLSCSum	
	Sentence	Comment	Sentence	Comment
RB (CCF) (S)	0.283	0.098	0.494	*0.308*
SVM	0.263†	0.089	0.440†	0.212†
SVMRank (S)	*0.294*	*0.104*	**0.515**	**0.318**
NMCF (S)	**0.312**	**0.117**	*0.505*	0.306

Again, ROUGE-scores from Table 4 validate the efficiency of our method, in which it is the best in two cases on SoLSCSum and second best in sentence selection on VSoLSCSum. On SoLSCSum, our method is the best following by SVM Ranking, which uses many sophisticated domain-dependent features to model sentence-comment relation. By contrast, our model is completely unsupervised learning, which is domain-independent. Other L2R methods such as RB CCF also obtain competitive results because they also use user posts to enrich information. SVM achieves the worst performance. A possible explanation is that it uses five simple features, e.g. sentence length [42]. Also, it does not exploit user posts.

[12] https://docs.scipy.org/doc/scipy-0.19.0/reference/generated/scipy.stats.ttest_ind.html

[13] https://sourceforge.net/p/lemur/wiki/RankLib

[14] https://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

Table 5: ROUGE-scores on two datasets. Bold is the best; *italic* is second best. Models with *S* use user posts in the summary process. LEAD is not used for comments. Values with † show that our model significantly outperforms with $p \leq 0.05$.

Method	SoLSCSum dataset				VSoLSCSum dataset			
	Sentence		Comment		Sentence		Comment	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
Lead-*m*	0.345	**0.322**	—	—	0.495†	0.420†	—	—
LexRank	0.327†	0.243†	*0.210*	0.115	0.506†	0.432†	0.348†	0.198†
SoviVote (S)	0.366†	0.297	0.150†	0.091†	0.482†	0.403†	0.353†	0.207†
cc-TAM (S)	0.306†	0.238†	0.054†	0.022†	0.448†	0.377†	0.301†	0.167†
HGRW (S)	*0.379*	0.204†	0.209	**0.122**	*0.570*	*0.479*	*0.454*	*0.298*
SoRTE-Inter Wing (S)	0.352†	0.277†	0.209	0.115	0.532†	0.463	0.443†	0.277
SoRTE-Dual Wing (S)	0.357†	0.280	0.180†	0.098	0.531†	0.457†	0.409†	0.234†
NMCF (S)	**0.387**	*0.312*	**0.218**	*0.117*	**0.590**	**0.505**	**0.483**	**0.306**

ROUGE-scores on DUC 2004. We finally confirm NMCF efficiency on DUC 2004. Our objective is to show the adaptation of our method on a standard dataset rather to obtain the best results on this dataset, which lacks user posts. Also, we would like to observe the margin between NMCF and state-of-the-art methods. DUC 2004 contains 50 topics, in which each topic has 10 articles and four references written by the human. Since this dataset has no comments, we adapted it for our model by using one-versus-all. We kept one article as the primary document and formed nine remaining articles as relevant information. We applied NMCF to each primary document to select two sentences, resulting 20 sentences in total. To select summaries, we employed a simple greedy algorithm, which is similar to the MMR strategy [6]. Sentences fewer than five words are first removed because they are short [8] and the rest is sorted in decreasing Cosine score as shown in Eq. (19).

$$score(s_i) = \frac{1}{|s_i|} \sum_{j=1}^{m} cos(s_i, s_j) \tag{19}$$

where m is all other sentences in the topic. We iteratively dequeue one sentence from the sorted list and append it into the summary if it is non-redundant (Cosine threshold = 0.75). The iteration stops if the summary reaches the length constraint.

We report three basic methods: PROB, LLR, MRW presented in [12] and three advanced methods: (i) MD-ILP, an abstractive ILP-based summarizer [2]; (ii) REGSUM, a regression-based model with hand-crated features [12]; and (iii) CRSum, a deep learning model based on sentence context [34] with ROUGE-1, 2, and 4 recall. Due to the setting of DUC, the evaluation uses the length constraint with the parameter ''b 665'' (665 bytes).

ROUGE-scores in Table 6 indicate that our method outputs competitive results on DUC 2004. It outperforms the three basic models: PROB, LLR, MRW. The ROUGE-scores of NMF are also promising, but our method is still better than NMF. However, there are rather gaps between NMCF with the state-of-the-art methods. For example, CRSum is the best in ROUGE-1 and REGSUM is the best in ROUGE-4. This is understandable that they are supervised learning. CRSum exploits the context surrounding a sentence in word and sentence level by using Bi-CNN [5] and LSTM [11], respectively. The final vector of a sentence is concatenated with several surface features such as its position. Also, REGSUM uses sophisticated hand-crafted features to estimate the importance of words, which

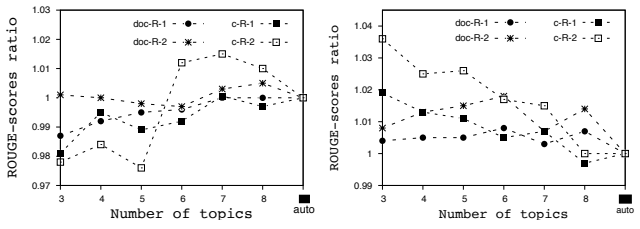
Table 6: ROUGE-scores on DUC 2004.

Method	ROUGE-1	ROUGE-2	ROUGE-4
PROB†	0.3514	0.0817	0.0106
LLR†	0.3460	0.0756	0.0083
MRW†	0.3578	0.0815	0.0099
MD-ILP†	—	**0.1199**	—
REGSUM†	*0.3857*	0.0975	**0.0160**
CRSum†	**0.3953**	*0.1060*	—
NMF	0.3557	0.0762	0.0107
NMCF	0.3734	0.0846	*0.0132*

can be used to measure the importance of sentences. By contrast, our model is unsupervised learning developed for social context summarization but not for multiple-document summarization without user posts. In another case, MD-ILP is the best in ROUGE-2 because it exploits the informativeness and linguistic quality with a set of constraints. However, adapting these methods for our task is still an open question. On the other hand, our method can be flexibly adapted to domains which include user posts (Figure 1 and Table 5) or not (Table 6) with very competitive results.

5.3 Topic Analysis

One of parameters in our model is the number of topics in creating the topic matrix W in Eqs. (6) and (7). We consider the sensitivity



(a) SoLSCSum **(b) VSoLSCSum**

Figure 2: ROUGE-scores on two datasets.

analysis of k in Eqs. (6) and (7) in two scenarios: (i) tuning topic number k and (ii) automatically selecting k in the optimization

Table 7: A summary example generated from the document 31^{st} on SoLSCSum; two summary sentences and comments are shown instead of six. Sentences with [+] means that they are also in the references; [-] means that they are not in the references.

NMCF Summarization	
Sentence selection	**Comment extraction**
[+]S1: Families of the victims of the Germanwings crash are considering filing a claim for damages in the United States if they cannot reach agreement with parent airline Lufthansa in Germany, a lawyer representing the families said on Sunday.	[+]C1: I'm sad for the people who are no longer on their life of this Germanwings strategy, but I don't know how much money relatives want for the flesh of who die on Germanwings case.
[-]S2: "If the airline is not prepared to do so, however, we will look seriously at making a claim in the United States," said Giemulla, adding that he was representing 21 families including those of the German school children who died.	[-]C2: The only ones who should be allowed to take this to a US court are the US citizens.
NMF Summarization	
Sentence selection	**Comment extraction**
[+]S1: Families of the victims of the Germanwings crash are considering filing a claim for damages in the United States if they cannot reach agreement with parent airline Lufthansa in Germany, a lawyer representing the families said on Sunday.	[-]C1: Julie to Ann (who is holding a cabbage): "I would have taken a ride with anyone except Peter Kramer, Freddie-maybe Joe.
[-]S2: Nearly half of the victims of the Germanwings Barcelona to Duesseldorf flight were German, with the remaining passengers hailing from a range of countries, including Spain, Australia and Argentina.	[-]C2: Don't understand why a US court would have authority over European citizens.

procedure. In the first scenario, we tuned k in [3, 8] with a jumping step = 1. The number of topics outside this rage is too small or large. In the second scenario, we choose $k = \frac{\#sentences}{2}$. More precisely, we normalize Figure 2 by dividing ROUGE-scores in each tuning point for those of k, which is automatically selected. Figure 2 shows that the topic number k affects the summarization quality. In Figure 2a, increasing k value improves the ROUGE-scores. Our model reaches a peak at $k = 7$ and 8, then its performance slightly decreases at $k = auto$. By contrast, in Figure 2b, the trend is reverse, in which increasing k reduces the ROUGE-scores. The gap between lower and upper bounds in both cases is insignificant (around 0.03) showing that changing k slightly changes the ROUGE-scores. Our model obtains the best results with $k = 5, 6, 7$. To balance on the two datasets, we select $k = 5$. Note that $k = \frac{\#sentences}{2}$ can also be used to avoid human-involvement in the summary process.

5.4 Normalization Observation

As mentioned, our model considers normalization L_1 or L_2 to avoid over-fitting. Table 8 shows the ROUGE-scores of the normalization.

Table 8: Norm observation. RG means ROUGE-scores.

Norm	SoLSCSum				VSoLSCSum			
	Sentence		Comment		Sentence		Comment	
	RG-1	RG-2	RG-1	RG-2	RG-1	RG-2	RG-2	RG-2
L_1	0.378	0.304	**0.222**	**0.125**	**0.594**	0.502	0.476	0.293
L_2	**0.387**	**0.312**	0.218	0.117	0.590	**0.505**	**0.483**	**0.306**

From Table 8 we can observe that normalization with L_1 shows its efficiency in some cases while L_2 is better in some metrics. For example, the ROUGE-scores of L_2 are mostly higher than those of L_1 on VSoLSCSum. On SoLSCSum, L_1 outperforms L_2 in comment extraction. In sentence selection the trend is reverse. From this observation, we selected L_2 as the normalization in our approach.

5.5 Case Study

From Table 7, we can observe that our model extracts one correct sentence and comment. The sentence locates in the second position. It contains much important information of the event "Germanwings crash families could seek damages in the U.S.: lawyer" such as the type of event ("Germanwings crash"), the situation (deal with the company for the damage). In many cases, reading S1 can provide enough salient information of this event. The comment C1 also reflects the content of this event. By reading C1, we can partly understand the event, in which the relatives of victims try to claim with the company for their money. Interestingly, C1 includes reader's opinion regarding the victims. However, topics mentioned in comments are usually diverse, then they bring the noise, such as C2. Our model also extracts incorrect sentences (S2 and C2). While S2 is relevant to the event, it is hard to conclude C2 belongs to this event. However, they include many words, which appear in important sentences, such as "airline", ''US'', or ''claim''. Also, as our expectation, the extracted sentences share many common words because we present the nature of sentences and comments in a share hidden topic-matrix. In this case, many common words help to improve the document representation.

NMF shares one correct sentence (S1) with our model. It is understandable since NMF is also competitive in summarizing documents. However, it selects the second one, which is relevant to the event, but not important at this moment because the event passed. This explains that our model outperforms NMF in sentence selection in Figure 1a. For comment extraction, interestingly, NMF extracts six incorrect comments (two of those are shown in Table 7). This explains the reason that our model significantly surpasses NMF in Figure 1b. Extracted sentences and comments share a few common words because it does not exploits the share of common topics.

6 CONCLUSION

This paper presents a model, which exploits user posts such as comments to enrich Web document summarization. The insight behind our model comes from the fact that sentences and comments share hidden topics in the form of common words or phrases. Our model captures the mutual information between sentences and comments by assuming they share hidden topics which can be found by a matrix co-factorization approach. The decomposition extracts salient sentences and comments, which have two characteristics: (i) reflecting document content and (ii) sharing common topics. Applying the model to the task of sentence extraction of single documents indicates that it can be viable alternative to extraction-based systems. ROUGE-scores on two datasets in English and Vietnamese in social context summarization and DUC 2004 confirm the efficiency of NMCF. The model achieves promising performance in an unsupervised fashion, without reference to any NLP tools (e.g. parsing) suggesting that it can be applied to unrestricted domains.

For future directions, an obvious next step is to investigate how the model works to other domains and text genres which include user posts. The document representation in §4.2.1 can also be presented by using semantic level such as word or sentence embedding.

ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant numbers 15K16048 and JP15K12094, and CREST, JST. This research was supported by UTEHY.T026.P1718.04. We would like to thank Dac Viet Lai for running the significant test and the anonymous referees for their valuable comments and helpful suggestions to improve our paper.

REFERENCES

[1] Einat Amitay and Cecile Paris. 2000. Automatically summarising web sites: is there a way around it?. In *Proceedings of the Ninth International Conference on Information and Knowledge Management, pp. 173-179. ACM.*

[2] Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. 2015. Multi-Document Abstractive Summarization Using ILP Based Multi-Sentence Compression. In *Twenty-Fourth International Joint Conference on Artificial Intelligence, pp. 1208-1214.*

[3] PB Baxendale. 1958. Man-made index for technical litterature - an experiment. *IBM J. Res. Dev., 2(4): 354-361* (1958).

[4] Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. 2015. Ranking with Recursive Neural Networks and Its Application to Multi-Document Summarization. In *AAAI, pp. 2153-2159.*

[5] Ziqiang Cao, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou, and Houfeng Wang. 2015. Learning Summary Prior Representation for Extractive Summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics ACL (2), pp. 829-833. Association for Computational Linguistics.*

[6] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 335-336. ACM.*

[7] Harold P. Edmundson. 1969. New Methods in Automatic Extracting. *Journal of the Association for Computing Machinery, 16(2), pp. 264-285* (1969).

[8] Gunes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research, 22, pp. 457-479* (2004).

[9] Wei Gao, Peng Li, and Kareem Darwish. 2012. Joint Topic Modeling for Event Summarization across News and Social Media Streams. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 1173-1182. ACM.*

[10] Yihong Gong and Xin Liu. 2001. Generic Text Summarization using Relevant Measure and Latent Semantic Analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 19-25. ACM.*

[11] Alex Graves, Abdel-Rahman Mohamed, and Geoffrey E. Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Acoustics, Speech and Signal processing (ICASSP), 2013 IEEE International Conference, pp. 6645-6649. IEEE.*

[12] Kai Hong and Ani Nenkova. 2014. Improving the Estimation of Word Importance for News Multi-Document Summarization. In *EACL, pp. 712-721.*

[13] Meishan Hu, Aixin Sun, and Ee-Peng Lim. 2008. Comments-Oriented Document Summarization: Understanding Document with Readers' Feedback. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 291-298. ACM.*

[14] Julian Kupiec, Jan O. Pedersen, and Francine Chen. 1995. A Trainable Document Summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 68-73. ACM.*

[15] Daniel D. Lee and H. Sebastian Seung. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature 401, no. 6755, pp. 788-791* (1999).

[16] Daniel D. Lee and H. Sebastian Seung. 2001. Algorithms for non-negative matrix factorization, In Advances in Neural Information Processing Systems, pp. 556-562. *Advances in Neural Information Processing Systems 13* (2001).

[17] Ju-Hong Lee, Sun Park, Chan-Min Ahn, and Daeho Kim. 2009. Automatic generic document summarization based on non-negative matrix factorization. *Information Processing & Management 45(1), pp. 20-34* (2009).

[18] Chen Li, Zhongyu Wei, Yang Liu, Yang Jin, and Fei Huang. 2016. Using Relevant Public Posts to Enhance News Article Summarization. In *COLING, pp. 557-566.*

[19] Chih-Jen Lin. 2007. Projected gradient methods for nonnegative matrix factorization. *Neural computation, 19(10), pp. 2756-2779* (2007).

[20] Chin-Yew Lin and Eduard H. Hovy. 2003. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, pp. 71-78. Association for Computational Linguistics.*

[21] Hui Lin and Jeff A. Bilmes. 2011, June. 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, pp. 510-520. Association for Computational Linguistics.*

[22] Yue Lu, ChengXiang Zhai, and Neel Sundaresan. 2009. Rated aspect summarization of short comments. In *Proceedings of the 18th International Conference on World Wide Web, pp. 131-140. ACM.*

[23] Hans P. Luhn. 1958. The Automatic Creation of Literature Abstracts. *IBM Journal of Research Development, 2(2), pp. 159-165* (1958).

[24] Preslav Nakov, Antonia Popova, and Plamen Mateev. 2001. Weight functions impact on LSA performance. In *EuroConference RANLP, pp. 187-193.*

[25] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: A Recurrent Neural Network based Sequence Model for Extractive Summarization of Documents. In *The Thirty-First AAAI Conference on Artificial Intelligence, pp. 3075-3081.*

[26] Ani Nenkova. 2005. Automatic text summarization of newswire: lessons learned from the document understanding conference. In *AAAI, vol. 5, pp. 1436-1441.*

[27] Minh-Tien Nguyen, Viet Dac Lai, Phong-Khac Do, Duc-Vu Tran, and Minh-Le Nguyen. 2016. VSoLSCSum: Building a Vietnamese Sentence-Comment Dataset for Social Context Summarization. In *The 12th Workshop on Asian Language Resources, pp. 38-48. Association for Computational Linguistics.*

[28] Minh-Tien Nguyen and Minh-Le Nguyen. 2016. SoRTESum: A Social Context Framework for Single-Document Summarization. In *European Conference on Information Retrieval, pp. 3-14. Springer International Publishing.*

[29] Minh-Tien Nguyen and Minh-Le Nguyen. 2017. Intra-relation or inter-relation?: Exploiting social information for Web document summarization. *Expert Systems with Applications 76, pp. 71-84* (2017).

[30] Minh-Tien Nguyen, Chien-Xuan Tran, Duc-Vu Tran, and Minh-Le Nguyen. 2016. SoLSCSum: A Linked Sentence-Comment Dataset for Social Context Summarization. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, pp. 2409-2412. ACM.*

[31] Miles Osborne. 2002. Using maximum entropy for sentence extraction. In *Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4, pp. 1-8. Association for Computational Linguistics.*

[32] Karolina Owczarzak, John M. Conroy, Hoa Trang Dang, and Ani Nenkova. 2012. An assessment of the accuracy of automatic evaluation in summarization. In *In Proceedings of Workshop on Evaluation Metrics and System Comparison for Automatic Summarization, pp. 1-9. Association for Computational Linguistics.*

[33] Sun Park, Ju-Hong Lee, Chan-Min Ahn, Jun Sik Hong, and Seok-Ju Chun. 2006. Query Based Summarization Using Non-negative Matrix Factorization. In *Knowledge-Based Intelligent Information and Engineering Systems, pp. 84-89. Springer Berlin/Heidelberg.*

[34] Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma, and Maarten de Rijke. 2017. Leveraging Contextual Sentence Relations for Extractive Summarization Using a Neural A ention Model. In *Proceedings of the 40th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 95-104. ACM.*

[35] Dou Shen, Jian-Tao Sun, Hua Li, Qiang Yang, and Zheng Chen. 2007. Document Summarization Using Conditional Random Fields. In *IJCAI, vol. 7, pp. 2862-2867.*

[36] Jian-Tao Sun, Dou Shen, Hua-Jun Zeng, Qiang Yang, Yuchang Lu, and Zheng Chen. 2005. Web-page summarization using clickthrough data. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 194-201. ACM.*

[37] Dingding Wang, Tao Li, Shenghuo Zhu, and Chris Ding. 2008. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 307-314. ACM.*

[38] Zhongyu Wei and Wei Gao. 2014. Utilizing Microblogs for Automatic News Highlights Extraction. In *COLING, pp. 872-883. Association for Computational Linguistics.*

[39] Zhongyu Wei and Wei Gao. 2015. Gibberish, Assistant, or Master?: Using Tweets Linking to News for Extractive Single-Document Summarization. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1003-1006. ACM.*

[40] Kristian Woodsend and Mirella Lapata. 2010. Automatic Generation of Story Highlights. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pp. 565-574. Association for Computational Linguistics.*

[41] Kristian Woodsend and Mirella Lapata. 2012. Multiple Aspect Summarization Using Integer Linear Programming. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 233-243. Association for Computational Linguistics.*

[42] Zi Yang, Keke Cai, Jie Tang, Li Zhang, Zhong Su, and Juanzi Li. 2011. Social Context Summarization. In *Proceedings of the 34th International SIGIR Conference on Research and Development in Information Retrieval, pp. 255-264. ACM.*