Statistical Text Summarization Using a Category-based Language Model on a Bootstrapping Framework

Hyoungil Jeong  
*Computer Science and Engineering, Sogang University, 35, Baekbeom-ro, Mapo-gu, Seoul, 04107, Korea*  
hijeong@gmail.com

Youngjoong Ko  
*Computer Engineering, Dong-A University, 37, Nakdong-daero 550beon-gil, Saha-gu, Busan, 49315, Korea*  
youngjoong.ko@gamil.com

Jungyun Seo  
*Computer Science and Engineering, Sogang University, 35, Baekbeom-ro, Mapo-gu, Seoul, 04107, Korea*  
seojy@sogang.ac.kr

Received (Day Month Year)  
Revised (Day Month Year)  
Accepted (Day Month Year)

Traditional text summarization systems have not used the category information of documents to be summarized. However, the estimated weights of each word can be often biased on small data such as a single document. Thus we proposed an effective feature-weighting method for document summarization that utilizes category information and solves the biased probability problem. The method uses a category-based smoothing method and a bootstrapping framework. As a result, in our experiments, our proposed summarization method achieves better performance than other statistical sentence-extraction methods.

*Keywords*: Text Summarization; Statistical Learning; Bootstrapping; Support Vector Machine.

1. Introduction

In recent years, the number of online news articles has increased significantly. The need to read these using smart devices continues to grow. This growth makes automatic text summarization more important\(^1\).

Automatic text summarization is distinguished from extractive summarization by statistical approaches and abstractive summarization by linguistic approaches\(^1\). Statistical extractive summarization based on sentence extraction is a simple yet strong summarization method. Although abstractive summarization can condense a document more effectively than the extractive method and produce a result that
is closer to what a human might generate, it is harder to develop, as it requires the use of natural language generation technology. Therefore, the generated summary sentences of an abstractive summarization are often grammatically incorrect and evaluation can become very difficult. On the other hand, statistical-extractive summarization selects salient sentences in a document, and thus the extracted sentences are usually grammatically correct.

In this paper, we propose a more effective sentence-extractive summarization method that is based on: 1) a language model with category information and 2) a bootstrapping approach. For the former, we focus on the bias of the estimating probability on small data from traditional statistical-extractive summarization approaches. They have generally used probabilistic information that is estimated in a document and its sentences. Unfortunately, a document and its sentences have few words. This causes the data sparseness problem. To overcome this problem, we apply a category-based language model to estimate the importance of the terms. This is an estimation of a word probabilistic model using document’s category and collection and also the document itself. To the best of our knowledge, category information has not been used in a fully automatic summarization except for guided summarization with expert-made, important-aspect templates for each category (for example, Category 'Accident and Natural Disasters', Aspect 'DAMAGES')\(^3\). For the latter, we bootstrap our extractive summarization method using a bootstrapping framework with a binary independence model (BIM). A BIM and its query-terms-weighting method is a probabilistic document retrieval technique\(^4\)\(^5\). This is useful in sentence-extractive summarization\(^6\). In the summarization method proposed by Jeong et al.\(^6\), they divide a document into relevant and irrelevant sentences and then select summary or non-summary sentences using a BIM. We improve this summarization method using a bootstrapping framework in which summary sentences are regarded as relevant sentences and non-summary sentences are regarded as irrelevant sentences in the previous iteration step. The proposed method achieved improvements of 12.56% and 12.81% in our two test datasets.

The remainder of this paper is organized as follows. In Section 2, we present the related work. In Section 3, we describe our proposed statistical document summarization method using a category-based language model and bootstrapping framework. In Section 4, we explain the experimental settings and results, and in Section 5 we conclude this paper.

2. Related work

Generally, there are two approaches to automatic summarization: extractive summarization and generative summarization. The extractive summarization works by selecting a subset of existing words, phrases, or sentences in the original text to form a summary. In contrast, the generative summarization, also known as abstractive summarization, builds an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what
a human might generate. Such a summary might contain words that are not explicitly presented in the original. Most of the researchers have concentrated, not on the generative summarization, but rather, the extractive summarization, in order to create a summary. This is primarily because of the limitation of natural language processing techniques in real application fields. The state-of-the-art generative methods are still quite weak, so most research has focused on extractive methods.

There have been many studies on the subject of the extractive summarization. Luhn pioneered the study of text summarization. He studied the most frequent words representing the most important concepts of a document. His representation abstracted the source document into a frequency table. Ko and Seo studied a statistical extractive summarization. They used the contextual information with bi-gram pseudo-sentences to overcome the data sparseness problem.

On the other views, there have been many studies on the minimization problem of redundancy in the summarization task. Carbonell and Goldstein pioneered the study of this problem, they presented the Maximal Marginal Relevance (MMR) technique that reduced redundancy while maintaining query relevance.

Some researchers were studied various abstractive summarization methods by the unit extraction. Takamura and Okumura studied the conceptual unit extraction by some decoding algorithms for summarization. Hirao et al. studied the rhetorical relations between the textual units for summarization. Filatova and Hatzivassiloglou proposed a formal model that can represents textual and conceptual units. However, as we mentioned in Section 1, these abstractive or unit-extractive summarization methods require the use of natural language generation technology, and also the results of an abstractive or unit-extractive summarization methods are often grammatically incorrect. In addition, evaluation can become very difficult because this may be evaluated manually. On the other hand, statistical-extractive summarization selects salient sentences in a document so the extracted sentences are usually grammatically correct. Thus, we focused on unsupervised learning based statistical text summarization in this paper. And it does not require heavy linguistic analyzers such as a syntactic parser and a semantic analyzer and only do a POS tagger.

In recent, many researchers tried to solve the summarization task by using deep neural net (DNN) technology. Rush et al. proposed a neural attention model. Cheng and Lapata studied a hierarchical document encoder and an attention-based extractor. Cao et al. developed a deep neural net summarizer called as AttSum. These DNN summarizers does not need hand-crafted features to train. These methods are promising. However, these methods need to train on large scale corpora of document-summary pairs. Unfortunately, in various environment, it is very difficult to gain these corpora yet.
3. Text Summarization Method Using Category Information

In this section, we explain our proposed text summarization method. The overview of our system is illustrated in Fig. 1.

Fig. 1. Proposed document summarization method.

This method is composed of the following five steps with a bootstrapping framework:

Step 1) Text Categorization: This is a general automatic document categorization that can classify input documents to predefined categories \(C_1 \sim C_N\) using a supervised machine learning method. We implement a general text categorizer by using a support vector machine (SVM) classifier\(^20\). We used the LIBSVM toolkit\(^21\). This is explained in Section 4.4.

Step 2) Relevant Sentence Selection: Relevant sentences are tentative summary sentences that are calculated by simple summarization methods including title similarity and sentence position. This is explained in Section 3.1.
Step 3) Term Relevance Estimation: The term relevance reflects how each term is related to relevant and non-relevant sentences. We estimate three types of relevance: document, category, and collection, in Section 3.2.

Step 4) Sentence Score Calculation: Each sentence is calculated by the term relevance of its own terms with a category-based language model using term distribution in document, category, and collection, in Section 3.3.

Step 5) bootstrapping Relevant Sentences: Some highly-scored sentences are assumed to be summary sentences. These summary sentences are considered as relevant sentences in Step 2. Until the summary sentences do not change, Steps 2 to 4 are repeated.

This approach is based on a document retrieval technique for relevance feedback. We use a blind relevance feedback (pseudo relevance feedback) and therefore, there is no restriction to the number of feedbacks. In comparison, the general relevance feedback method has difficulty with multiple iterations (feedbacks) because they come from human users. Until the summary sentences do not change, the steps are repeated based on the assumption that the summary sentences are relevant sentences and the non-summary sentences are irrelevant sentences. This can be noted in our bootstrapping framework.

3.1. Initial Relevant Sentence Selection

The title of the article and position of sentences are regarded as salient criteria for document summarization. The title of a news article often constitutes an abstraction of the entire content of the article itself. Sentences similar to the title are candidates to be salient sentences. News articles also usually have a deductive structure, that is, the beginning sentences are often considered as salient sentences. The beginning sentences of text is not guaranteed that it is salient, but it can be used initial relevant sentences as a tentative summary. This fact can reflect on the summarization by the Relevant Score from Eq. (1). We can separate all the sentences of a document into relevant and irrelevant sentences according to similarity between title and each sentence and the position of each sentence. This relevant score \( (Rel) \) is estimated using the similarity with the title \( (Sim) \) and position score \( (Pos) \) in Eq. (1) as follows:

\[
Rel(S_i) = Sim(S_i, T) + Pos(S_i) \tag{1}
\]

where \( S_i \) is the \( i \)-th sentence and \( T \) is the title of a document \( d \).

Similarity \( (Sim) \) is calculated by Eq. (2).

\[
Sim(S_i, T) = \frac{(S_i \cdot T)}{\max_{s_j \in d} (S_j \cdot T)} = \frac{\sum_{t \in S_i \cap T} tf(S_i, t) \times tf(t, T)}{\max_{s_j \in d} (\sum_{t \in S_j \cap T} tf(t, S_j) \times tf(t, T))} \tag{2}
\]
where \( tf(t, S_i) \) is the term frequency of term \( t \) in \( S_i \).

Eq. (2) is the similarity of \( S_i \) and \( T \), and also this is a normalized dot-product of \( S_i \) and \( T \). This is calculated on common terms on \( S_i \) and \( T \). If there are no common terms, it has zero similarity. In Eq. (2), the denominator \( \max_{j \in d}(S_j \cdot T) \) means to pick out the sentence \( j \) in the document \( d \) with the maximum value of \( (S_i \cdot T) \). By this denominator, each sentence-title similar score \( (S_i \cdot T) \) is normalized to less 1.0 where the sentence with maximum has the similarity score 1.0.

Position \( (Pos) \) is calculated by Eq. (3).

\[
Pos(S_i) = 1.0 - \frac{i - 1}{N}
\]

where \( N \) is the number of sentences in the document \( d \).

In the \( Rel \) ranking, we set the relevant sentences as the top 30 percent and consider the remainder as irrelevant sentences.

In this paper, the title and location method is used on initial relevant sentences selection. This is an important step because the quality of initialization affect substantially on the final result. However, this initialization is not the core of our approach. To all intents, the core aspects are the category-based language model and the bootstrapping framework. The initial relevant sentences are roughly extracted as a tentative summary, and these are finely tuned in the following steps.

### 3.2. Term Relevance Estimation

The term relevance \( (TR) \) reflects how each term is related to the relevant or irrelevant sentences. The term relevance can be calculated by Eq. (4).

\[
TR_d(t) = \log \frac{p_d \times (1 - q_d)}{(1 - p_d) \times q_d} = \log \left( \frac{(r_d + 0.5) \times (1 - (s_d + 0.5))}{(1 - r_d + 0.5) \times s_d + 0.5} \right) = \log \left( \frac{r_d + 0.5}{R_d - r_d + 0.5} \right) \times (s_d + 0.5)
\]

where \( p_d \) and \( q_d \) are the probabilities that term \( t \) appears in relevant and irrelevant sentences in a document \( d \), respectively; \( R_d \) and \( S_d \) are the number of relevant sentences and irrelevant sentences in document \( d \), respectively, \( (R_d + S_d = |d|) \); \( r_d \) and \( s_d \) are the number of relevant and irrelevant sentences that include term \( t \) in a document \( d \), respectively; 0.5 is a naive smoothing factor to avoid zero-denominator or log-zero.

This is based on the retrieval status value in the query terms weighting method of the binary independence model, introduced by Yu and Salton and coined by Robertson and Jones\textsuperscript{4,5}. The query terms weighting method of the binary independence model was developed for the traditional information retrieval task, in order
to make the estimation of document-query similarity probability. We adopted the document summarization task. This weighting method, $TR$, is defined by the log-odds ratio between relative and irrelative probability. If $TR$ has a positive large value, term $t$ is highly related with relevant sentences. On the contrary, if $TR$ has a negative large value, term $t$ is highly related with irrelevant sentences. If $TR$ is close to zero, then term $t$ is not related to either relevant or irrelevant sentences.

$TR_d$ is dependent on each document. In addition, there exist terms that are dependent on category. Thus, $TR_d$ can be extended to category $c$ as the smoothing factor in order to extract more information. This can be denoted by $TR_c$ and can be calculated by Eq. (5).

$$TR_c(t) = \log \left( \frac{(r_c + 0.5) \times (S_c - s_c + 0.5)}{(R_c - r_c + 0.5) \times (s_c + 0.5)} \right)$$

where $R_c$ and $S_c$ are the number of relevant sentences and irrelevant sentences in a category $c$, respectively, $(R_c + S_c = |c|)$; $r_c$ and $s_c$ are the number of relevant and irrelevant sentences that include term $t$ in a category $c$, respectively.

Similar to Eq. (5), there exist terms that are category independent. $TR_c$ can be extended to the collection of dataset $COL$ as the smoothing factor in order to extract more information. This can be denoted as $TR_{COL}$ and calculated by Eq. (6).

$$TR_{COL}(t) = \log \left( \frac{(r_{COL} + 0.5) \times (S_{COL} - s_{COL} + 0.5)}{(R_{COL} - r_{COL} + 0.5) \times (s_{COL} + 0.5)} \right)$$

where $R_{COL}$ and $S_{COL}$ are the number of relevant sentences and irrelevant sentences in the collection of a dataset $COL$, respectively, $(R_{COL} + S_{COL} = |COL|)$; $r_{COL}$ and $s_{COL}$ are the number of relevant and irrelevant sentences that include term $t$ in the collection of a dataset $COL$, respectively.

### 3.3. Sentence Score Calculation

In single documents or sentences, each word can be biased to estimate their own weight because they appeared in a few times. To overcome this problem, language-smoothing methods have been used by many researchers in the field of information retrieval and natural language processing. We use these smoothing methods for each sentence. We calculate the Information Score ($IS$) for each sentence using TR. This score is calculated by the document-term relevance, internal category-term relevance, and collection-term relevance. $IS$ is calculated by Eq. (7).
Then, \( IS(S_i) \) is a ratio for the document and category information score and \( \beta \) is a ratio for the category and collection information score. These are calculated in Eq. (8).

\[
\alpha = \frac{|d|}{|d| + \mu_1}, \quad \beta = \frac{|c|}{|c| + \mu_2}
\]

where \(|d|\) is the number of terms in a document \(d\) and the \(|c|\) is the number of terms in a category \(c\), in addition, \(\mu_1\) is the average number of terms in the document, and \(\mu_2\) is the average number of terms in the category.

\(\mu_1\) and \(\mu_2\) are similar to the familiar smoothing parameters of the cluster-based language model\(^{2}\) that originated in the Dirichlet parameter as a smoothing method for language modeling in the field of the information retrieval. The traditional Dirichlet prior-smoothing method uses log probability; the term relevance uses the log-odds ratios. Thus, the traditional Dirichlet prior-smoothing method cannot use the term relevance directly. We propose, therefore, this smoothing method using the Dirichlet parameter.

This needs a document categorization. Unfortunately, their performances are not ideal. Therefore, we use a latent class model\(^{25}\) for category \(c\). This is presented in Eq. (9).

\[
IS(S_i) = \alpha \times \sum_{t \in S_i} TR_d(t) + \\
(1.0 - \alpha) \times \beta \times \sum_{t \in S_i} TR_c(t) + \\
(1.0 - \alpha) \times (1.0 - \beta) \times \sum_{t \in S_i} TR_{COL}(t)
\]

where the probability of category \(c\) given a document \(d\) \(P(c|d)\) is estimated by the pairwise coupling method that combines together all pairwise comparisons for each pair of classes\(^{26}\).

As a result, the high-ranked sentences (top 30\%) are repeatedly selected as a summary of each news article by the \(IS\) function.

The small frequency of each term in single sentence or document often makes biased weights of each term. The \(IS\) function can smooth the frequency on relevant or
irrelevant sentences by using frequencies from their related categories with a large volume; the related categories are selected and the degree of a related category is reflected by $P(c|d)$ in Eq. (9). Thus this can help estimate the unbiased weights of each term.

4. Experiments

4.1. Datasets

We need datasets annotated with category and summary information labels to evaluate the performance of our automatic document summarization using category information. Since these datasets were not available, we reconstructed two datasets by annotating the category or summary information.

One dataset was collected from the AbleNews dataset from the year 2012\textsuperscript{27}. This is a Korean online newspaper site with category information. For this dataset, we created an extractive summary for each article. The other one was the KORDIC dataset\textsuperscript{28}. This is a well-known Korean news-summarization test dataset. We annotated the category information for each article of this dataset. These datasets are presented in Table 1.

The following steps are done for annotation. Three annotators and one coordinator are employed for our annotation task. Each annotator suggests golden-standard summary sentences and golden-standard category of a document. If a summary sentence and a category are suggested as golden-standard by two or three annotators, they are decided as the golden-standard summary sentence and the category. If annotators’ suggestion has a divergence, a coordinator decides the golden-standard sentences or a category.

<table>
<thead>
<tr>
<th>Category c</th>
<th># of doc.</th>
<th>Note</th>
<th># of doc.</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>227</td>
<td>Politics</td>
<td>111</td>
<td>Politics</td>
</tr>
<tr>
<td>C2</td>
<td>311</td>
<td>Society</td>
<td>249</td>
<td>Society</td>
</tr>
<tr>
<td>C3</td>
<td>132</td>
<td>Economy</td>
<td>113</td>
<td>Economy</td>
</tr>
<tr>
<td>C4</td>
<td>81</td>
<td>Education</td>
<td>24</td>
<td>Education</td>
</tr>
<tr>
<td>C5</td>
<td>74</td>
<td>Life</td>
<td>105</td>
<td>Life</td>
</tr>
<tr>
<td>C6</td>
<td>157</td>
<td>Welfare/Health</td>
<td>209</td>
<td>World</td>
</tr>
<tr>
<td>C7</td>
<td>90</td>
<td>Woman/Children</td>
<td>34</td>
<td>Ecology</td>
</tr>
<tr>
<td>C8</td>
<td>183</td>
<td>Culture</td>
<td>52</td>
<td>IT/Science</td>
</tr>
<tr>
<td>C9</td>
<td>137</td>
<td>Person/Association</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>COL</td>
<td>1392</td>
<td></td>
<td>897</td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Description of datasets.
4.2. Evaluation Measures

The performance of the summarization and categorization was evaluated using precision, recall, and F1-measure.

In the evaluation of the summarization, precision is a ratio of the number of suggested golden-standard summary sentences to the total suggested summary sentences in a document; recall is the number of suggested golden-standard summary sentences to the total golden-standard summary sentences in a document; and F1-measure is the harmonic mean between the precision and recall in a document. For the entire dataset, the arithmetic mean of the F1-measures for each document was used.

In the evaluation of categorization, precision is a ratio of the number of suggested golden-standard document classes to the total suggested document classes in the category; recall is the ratio of the number of suggested golden-standard document classes to the total golden-standard document classes in the category, and F1-measure is the harmonic mean between precision and recall in a category. For the entire dataset, an arithmetic mean of F1-measures for each category is used.

4.3. Compared Methods

We made use of six other extractive summarizations for comparing summarization results, including Title Method\textsuperscript{22}, Location Method\textsuperscript{22}, Frequency Method\textsuperscript{23}, Aggregation Method\textsuperscript{8}, Pseudo Relevance Feedback Method\textsuperscript{6}, and Contextual Information Method\textsuperscript{12}. These statistical sentence-extracting methods are summarized as follows:

1) Title Method: Sentences that have high cosine-similarity with the title are extracted as a summary.
2) Location Method: The beginning sentences of an article are extracted as a summary.
3) Frequency Method: Sentences that have a high sum of TF-IDF values of terms are extracted as a summary.
4) Aggregation Method: Sentences that have high sum of similarity to other sentences are extracted as a summary.
5) Pseudo Relevance Feedback Method: Sentences that have high sum of relevance feedback of terms are extracted as a summary.
6) Contextual Information Method: Sentences that have maximum contextual information are extracted as a summary.

4.4. Preliminary Experiment: Setup for the Text Categorization

We implemented a general text categorizer by using the linear kernel of the LIBSVM toolkit\textsuperscript{21} by using lexical feature as morphemes of each document. We set to the multi-class SVM with the one-versus-all method, and also we used the five-folds cross validation for a fair evaluation. We did experiments to compare the three feature-weighting schemes: the binary weight, the TF weight, and the TF-IDF weight\textsuperscript{29,30}. Their performances are represented in Fig. 2.
In Fig. 2, the X-axis is precision, recall, and F1-measure macro-averaged for each category; the Y-axis is the performance of each.

From the results of this experiment, the TF-IDF weighting scheme was selected because it showed a better F1-measure than the other schemes. In fact, this text categorization is not state-of-the-art, but this basic categorization could approach the highest summarization despite its error as a categorical output.

4.5. Experimental Results of Summarization

The performance (F1-measure) of the first proposed document summarization using $TR_d$, $TR_c$, and $TR_{COL}$ are shown in Table 2.

<table>
<thead>
<tr>
<th>Used Information</th>
<th>AbleNews</th>
<th>KORDIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only Document ($TR_d; \alpha = 1.0$; baseline)</td>
<td>0.581</td>
<td>0.531</td>
</tr>
<tr>
<td>Document and Category ($TR_d, TR_c; \beta = 1.0$)</td>
<td>0.622 (+7.06%)*</td>
<td>0.560 (+5.46%)*</td>
</tr>
<tr>
<td>All ($TR_d, TR_c, TR_{COL}$)</td>
<td>0.639 (+9.98%)*</td>
<td>0.581 (+9.42%)*</td>
</tr>
</tbody>
</table>

(* significant; $p < 0.05$)
In the case of using $TR_d$, $TR_c$, and $TR_{COL}$, we obtained higher performances in the two datasets than when only using $TR_d$. These are improved performances, 9.98% in the AbleNews and 9.42% in the KORDIC datasets, compared to the baseline relatively. These results are significantly improved less than 0.05 of p-value by paired t-test.

The performance by the number of pseudo relevance feedbacks (iterations) in our bootstrapping framework is shown in Fig. 3.

![Fig. 3. Performances by bootstrapping framework.](image)

In Fig. 3, the X-axis is the number of iterations and the Y-axis is the $F1$-measure. They converge at five or six iterations to 0.654 and 0.599, indicating relatively improved performances of 12.56% and 12.81% more than the baseline, in the AbleNews and KORDIC datasets, respectively.

Table 3 shows the comparison results with other document summarization methods.

As can be seen from Table 3, the proposed method achieved the best performance among all the document summarization methods. The performance of the proposed system on the AbleNews dataset is more remarkable than that on the KORDIC dataset. This could be related to the category-based smoothing and the fact that the size of each category of the AbleNews is more evenly distributed and larger than the KORDIC dataset.
### Table 3. Performances of the document summarization.

<table>
<thead>
<tr>
<th>Summarization Method</th>
<th>AbleNews</th>
<th>KORDIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proposed System</td>
<td>0.654</td>
<td>0.599</td>
</tr>
<tr>
<td>Title Method</td>
<td>0.475</td>
<td>0.488</td>
</tr>
<tr>
<td>Location Method</td>
<td>0.565</td>
<td>0.494</td>
</tr>
<tr>
<td>Frequency Method</td>
<td>0.399</td>
<td>0.378</td>
</tr>
<tr>
<td>Aggregation Method</td>
<td>0.388</td>
<td>0.415</td>
</tr>
<tr>
<td>Pseudo Relevance Feedback Method</td>
<td>0.568</td>
<td>0.511</td>
</tr>
<tr>
<td>Contextual Information Method</td>
<td>0.564</td>
<td>0.553</td>
</tr>
</tbody>
</table>

The proposed summarization can enhance the title similarity and the sentence-location methods. In general, the pseudo-relevance feedback method and the binary independence model have been used for the task of information retrieval, and these methods could strongly enhance a poor retrieval system. In this paper, we proved that the pseudo-relevance feedback method can be well adapted to sentence-extractive summarization.

### 5. Conclusion

In this paper, we proposed a statistical text summarization method that utilizes the category information of each document and a bootstrapping framework. It uses the binary independent model’s query weighting method and a category-based language model smoothing method to solve the data sparseness problem and to improve the performance using category information.

In the experimental results, our proposed summarization method showed the best performance among other statistical summarization methods by sentence-extraction. It also indicated that category information usage is effective in text summarization. Moreover, the proposed method is independent of language, i.e., the statistical methods that are used in the proposed method can be applied to any other language. This is another advantage of the proposed method.

### References

19. Z. Cao, W. Li, S. Li, F. Wei, and Y. Li, AttSum: Joint Learning of Focusing and Summarization with Neural Attention, in *Proc. 26th Int. Conf. on COLING* (2016) pp. 547–556