DOCUMENTS AS A BAG OF MAXIMAL SUBSTRINGS
An Unsupervised Feature Extraction for Document Clustering

Tomonari Masada, Yuichiro Shibata and Kiyoshi Oguri
Graduate School of Engineering, Nagasaki University, 1-14 Bunkyo-machi, Nagasaki-shi, Nagasaki, 8528521, Japan

Keywords: Maximal substrings, Document clustering, Suffix array, Bayesian modeling.

Abstract: This paper provides experimental results showing how we can use maximal substrings as elementary features in document clustering. We extract maximal substrings, i.e., the substrings each giving a smaller number of occurrences even after adding only one character at its head or tail, from the given document set and represent each document as a bag of maximal substrings after reducing the variety of maximal substrings by a simple frequency-based selection. This extraction can be done in an unsupervised manner. Our experiment aims to compare bag of maximal substrings representation with bag of words representation in document clustering. For clustering documents, we utilize Dirichlet compound multinomials, a Bayesian version of multinomial mixtures, and measure the results by F-score. Our experiment showed that maximal substrings were as effective as words extracted by a dictionary-based morphological analysis for Korean documents. For Chinese documents, maximal substrings were not so effective as words extracted by a supervised segmentation based on conditional random fields. However, one fourth of the clustering results given by bag of maximal substrings representation achieved F-scores better than the mean F-score given by bag of words representation. It can be said that the use of maximal substrings achieved an acceptable performance in document clustering.

1 INTRODUCTION

Recently, researchers propose a wide variety of methods envisioning large scale data mining, where documents originating from SNS environments or DNA/RNA sequences provided by next generation sequencing are a typical target of their proposals. Many of the methods adopt an *unsupervised* approach, because it is difficult to prepare a sufficient amount of hand-maintained training data for a *supervised* learning when test data is of quite large scale.

This paper focuses on text mining, where we can use various unsupervised methods, e.g. document clustering (Nigam et al., 2000), topic extraction (Blei et al., 2003), topical trend analysis (Wang and McCallum, 2006), etc, each based on an elaborated document modeling. However, these unsupervised methods assume that each document is represented as a *bag of words*, i.e., as an unordered collection of words. Therefore, we should first extract elementary features that can be called words.

For some languages, e.g. English, French, German, etc, we can obtain words by a simple heuristics called *stemming*. However, for many languages, e.g. Japanese, Chinese, Korean, etc, it is far from a trivial task to extract words. Japanese and Chinese sentences contain no white spaces and thus give no boundaries between the words. While Korean sentences contain white spaces, each string separated by a white space often consists of multiple words (Choi et al., 2009).

Many of the existing word extraction methods require a well-maintained dictionary and/or a well-trained data model of character sequences. Further, such extraction methods often presuppose the availability of a sufficient amount of training data to which supervised signals (e.g. 0/1 labels giving word boundaries, grammatical categories of words, etc) are assigned by hand. Therefore, any mining method sitting on such *supervised* feature extraction methods may show difficulty in scaling up to larger datasets even when the mining method itself is an unsupervised one.

This paper shows how we can use *maximal substrings* (Okanohara and Tsujii, 2009) as elementary features of documents. One important characteristic of maximal substrings is that they can be obtained in a totally *unsupervised* manner.

In this paper, we compare bag of maximal substrings representation with bag of words representation in *document clustering*. To be precise, we compare the quality of document clusters obtained by us-

Masada T., Shibata Y. and Oguri K..
DOCUMENTS AS A BAG OF MAXIMAL SUBSTRINGS - An Unsupervised Feature Extraction for Document Clustering.
DOI: 10.5220/0003403300050013
In *Proceedings of the 13th International Conference on Enterprise Information Systems* (ICEIS-2011), pages 5-13
ISBN: 978-989-8425-53-9

ing maximal substrings as document features with the quality of document clusters obtained by using words as document features. We compute document clusters by applying the same clustering method to the same document set. In our comparison, only the document representation is different.

As far as we know, this paper is the first one that gives a quantitative comparison of bag of maximal substrings representation with bag of words representation in document clustering. While Chumwatana et al. conduct a similar experiment with respect to Thai documents (Chumwatana et al., 2010), they fail to give reliable evaluation, because their datasets consist of only tens of documents. Further, they do not compare bag of maximal substrings representation with bag of words representation.

We conducted document clustering on tens of thousands of Korean and Chinese newswire articles. To compare with maximal substrings, we extracted words by applying a dictionary-based morphological analyzer (Gang, 2009) to Korean documents and by applying a word segmenter implemented by us based on linear conditional random fields (CRF) (Sutton and McCallum, 2007) to Chinese documents. The former extraction method presupposes that we have a well-maintained dictionary, and the latter presupposes that we have a sufficient amount of training data.

Our experiment will provide the following important observations:

- For Korean documents, maximal substrings are as effective as words extracted by the dictionary-based morphological analyzer.
- For Chinese documents, maximal substrings are not so effective as words extracted by the CRF-based word segmenter. However, the performance achieved with maximal substrings is acceptable.

The rest of the paper is organized as follows. Section 2 gives previous works related to the extraction of elementary features from documents. Section 3 describes how we use maximal substrings as elementary features in Bayesian document clustering. Section 4 includes the procedure and the results of our evaluation experiment. Section 5 concludes the paper with discussions and future work.

2 PREVIOUS WORKS

Most text mining methods require word extraction as a preprocessing of documents. We have a relatively simple heuristics called stemming for English, French, German, etc. However, it is far from a trivial task to extract elementary linguistic features that can be called words for many languages, e.g. Japanese, Chinese, Korean, etc.

Word extraction can be conducted, for example, by analyzing language-specific grammatical structures with a well-maintained dictionary (Gang, 2009), or by labeling sequences with an elaborated probabilistic model whose parameters are in advance optimized with respect to hand-prepared training data (Tseng et al., 2005). However, recent research trends point to increasing need for large scale data mining. Therefore, intensive use of supervised word extraction becomes less realistic, because it is difficult to prepare training data of sufficient size and quality.

Actually, we already have important results for unsupervised feature extraction from documents.

Poon et al. (Poon et al., 2009) propose an unsupervised word segmentation by using log-linear models, often adopted for supervised word segmentation, in an unsupervised learning framework. However, when computing the expected count that is required in learning process, the authors exhaustively enumerate all segmentation patterns. Consequently, this approach is only applicable to the languages whose sentences are given as a sequence of short strings separated by white spaces (e.g. Arabic and Hebrew), because the total number of segmentation patterns are not so large for each short strings. That is, this approach will show an extreme inefficiency in execution time for the languages whose sentences are given with no white spaces (e.g. Chinese and Japanese).

Mochihashi et al. (Mochihashi et al., 2009) provide a sophisticated Bayesian probabilistic model for segmenting given sentences into words in a totally unsupervised manner. The authors improve the generative model of Teh (Teh, 2006) and utilize it for modeling both character *n*-grams and word *n*-grams. The proposed model can cope with the data containing so-called out-of-vocabulary words, because the generative model of character *n*-grams serves as a new word generator for that of word *n*-grams. However, highly complicated sampling procedure, including MCMC for the nested *n*-gram models and segmentation sampling by an extended forward-backward algorithm, may encounter an efficiency problem when we try to implement this procedure, though the proposed model is well designed enough to prevent any exhaustive enumeration of segmentation candidates.

Okanohara et al. (Okanohara and Tsujii, 2009) propose an unsupervised method from a completely different angle. The authors extract *maximal substrings*, i.e., the substrings each giving a smaller number of occurrences even after adding only one character at its head or tail, as elementary features. This extraction can be efficiently implemented based

on the works related to suffix array and Burrows-Wheeler transform (Kasai et al., 2001; Abouelhoda et al., 2002; Navarro and Makinen, 2007; Nong et al., 2008). While Zhang et al. (Zhang and Lee, 2006) also provide a method for extracting a special set of substrings, this is not the set of maximal substrings. Further, their method has many control parameters and thus seems based on a more heuristic intuition when compared with the extraction of maximal substrings.

In this paper, we adopt maximal substrings as elementary features of documents by following the line of Okanohara et al. and check the effectiveness in *document clustering*, because both previous works (Okanohara and Tsujii, 2009; Zhang and Lee, 2006) try to prove the effectiveness of extracted substrings in *document classification*.

We can also find previous works using maximality of substrings for document clustering. Zhang et al. (Zhang and Dong, 2004) present a Chinese document clustering method by using maximal substrings as elementary features. However, the authors give no quantitative evaluation. Especially, maximal substrings are not compared with elementary features extracted by an elaborated supervised method. While Li et al. (Li et al., 2008) also propose a document clustering based on the maximality of subsequences, the authors focus not on character sequences, but on word sequences. Further, the proposed method utilizes WordNet, i.e., an external knowledge base, for reducing the variety of maximal subsequences and thus is not an unsupervised method.

In this paper, we would like to show what kind of effectiveness maximal substrings can provide in document clustering when we only use a frequency-based selection method for reducing the variety of substrings and use no external knowledge base.

3 DOCUMENT CLUSTERING WITH MAXIMAL SUBSTRINGS

3.1 Extracting Maximal Substrings

Maximal substrings are defined to be a substring whose number of occurrences is reduced even by adding only one character to its head or tail. We discuss more formally below.

We assume that we have a string S of length $l(S)$ over a lexicographically ordered character set Σ. In addition, we assume that a special character \$, called sentinel, is attached at the tail of S, i.e., $S[l(S)] = \$$. The sentinel \$ does not appear in the given original string and is smaller than all other characters in lexicographical order.

For a pair of strings S and T over Σ, we define $Pos(S,T)$ by $Pos(S,T) \equiv \{i : S[i+j-1] = T[j] \text{ for } j = 1,\ldots,l(T)\}$, i.e., the set of all occurrence positions of T in S. We denote the nth smallest element in $Pos(S,T)$ by $pos_n(S,T)$. Further, we define $Rel(S,T)$ by $Rel(S,T) \equiv \{(n, pos_n(S,T) - pos_1(S,T)) : n = 1,\ldots,|Pos(S,T)|\}$. $Rel(S,T)$ is the set of all occurrence positions relative to the first smallest occurrence position. Then, T is a *maximal substring* of S when the following conditions hold:

1. $|Pos(S,T)| > 1$;
2. $Rel(S,T) \neq Rel(S,T')$ for any T' s.t. $l(T') = l(T)+1$ and $T[j] = T'[j], j = 1,\ldots,l(T)$; and
3. $Rel(S,T) \neq Rel(S,T')$ for any T' s.t. $l(T') = l(T)+1$ and $T[j] = T'[j+1], j = 1,\ldots,l(T)$.

The last condition corresponds to "left expansion" discussed by Okanohara et al. (Okanohara and Tsujii, 2009).

When we extract maximal substrings from a document set, we first concatenate all documents by inserting a special character, which does not appear in the given document set, between the documents. The concatenation order is irrelevant to our discussion. We put a sentinel at the tail of the resulting string and obtain a string S from which we extract maximal substrings. We can efficiently extract all maximal substrings from S in time proportional to $l(S)$ (Okanohara and Tsujii, 2009).

The number of different maximal substrings is in general far larger than that of different words obtained by morphological analysis or by word segmentation. Therefore, we remove maximal substrings containing special characters put between the documents. Further, when the target language provides its sentences with white spaces, delimiters (e.g. comma, period, question mark, etc), and other functional characters (e.g. parentheses, hyphen, center dot, etc), we remove maximal substrings containing such characters.

Even after the above reduction, we still have a large number of different maximal substrings. Therefore, we propose a simple frequency-based strategy for reducing the variety of maximal substrings. We only use the following two parameters: the lowest and the highest frequencies of maximal substrings.

To be precise, we remove all maximal substrings whose frequencies are less than the threshold n_L and also remove all maximal substrings whose frequencies are more than the threshold n_H. We directly specify n_L as 10, 20, 50, etc. On the other hand, we specify n_H through the equation $n_H = c_H \times n_1$, where n_1 is the frequency of the most frequent maximal substring and c_H is a real value. We feel difficulty in specifying n_H

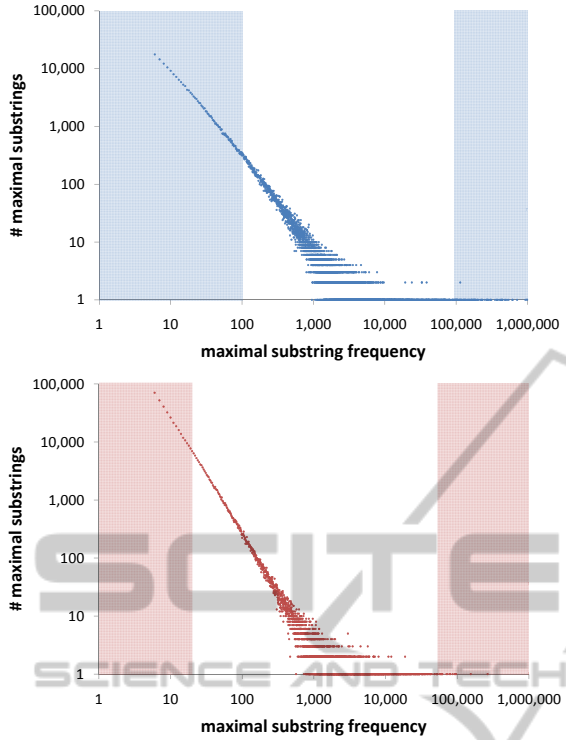

Figure 1: Plot of the number of maximal substrings for each different frequency. The top panel (resp. bottom panel) shows the statistics of maximal substrings extracted from Korean (resp. Chinese) newswire articles used in our evaluation experiment. For example, when we have 800 different maximal substrings each occurring 100 times in the document set, a marker is placed at $(100, 800)$ in the chart.

directly, because the choice of n_H heavily depends on the size of the dataset. However, it can be regarded that n_1 scales with the size of the dataset. Therefore, we specify n_H by multiplying a factor c_H to n_1.

Figure 1 shows how many different maximal substrings we have at each frequency for the two datasets used in our experiment. The top panel gives the statistics of maximal substrings for the Korean newswire article set, and the bottom panel gives the statistics for the Chinese article set. The horizontal axis represents the frequency of maximal substrings, and the vertical axis represents the number of different maximal substrings having the same frequency. For example, when we have 800 different maximal substrings each occurring at 100 positions in the document set, a marker is placed at $(100, 800)$ in the chart. The data plot range starts from five on the horizontal axis, because the maximal substrings of frequency less than five were noisy strings and were thus removed. We can observe that the distribution of the number of different maximal substrings roughly follows Zipf's law.

The two shaded areas of each chart in Figure 1 show the frequency intervals where the corresponding maximal substrings are removed by our selection method. In other words, the unshaded area of each chart ranges from n_L to n_H on the horizontal axis and thus shows the frequency interval where the corresponding maximal substrings are used in document clustering. Figure 1 displays the frequency interval giving the best evaluation result for each dataset. For the chart in the top panel, n_L is set to 100 and n_H to $97,879$, where n_H is determined by setting c_H to 0.1 with respect to $n_1 = 978,789$. For the chart in the bottom panel, n_L is set to 20 and n_H to $53,051$, where n_H is determined by setting c_H to 0.2 with respect to $n_1 = 265,254$. Each setting gave the best result as we will discuss in Section 4.

3.2 Bayesian Document Clustering

When documents are represented as a bag of elementary features, multinomial distribution (Nigam et al., 2000) is a natural choice for document modeling, because we can represent each document as a frequency histogram of elementary features.

However, it is often discussed from a Bayesian view point that multinomial distributions are likely to overfit to *sparse* data. The term "sparse" means that the number of different features appearing in each document is far less than the total number of different features observable in the entire document set. This tendency becomes more apparent when we use maximal substrings as document features, because the number of maximal substrings extracted from a document set is in general far larger than that of words extracted by an elaborated word segmentation method.

Therefore, we use Dirichlet compound multinomials (DCM) (Madsen et al., 2005) as our document model to avoid overfitting. We assume that the number of clusters is K. DCM has K multinomial distributions, each of which models a word frequency distribution for a different document cluster. Further, DCM applies a Dirichlet prior distribution to each of the K multinomials. DCM can effectively avoid overfitting with these K Dirichlet prior distributions.

Here we prepare notations for discussions. We assume that the given document set contains J documents and that W different words (or maximal substrings) can be observed in the document set. Let c_{jw} be the number of occurrences of the wth word (or maximal substring) in the jth document. Let α_{kw}, $k = 1, \ldots, K$, $w = 1, \ldots, W$ be the hyperparameters of the Dirichlet priors. The posterior probability that the jth document belongs to the kth cluster is denoted by p_{jk}. Note that $\sum_k p_{jk} = 1$. We define $\alpha_k \equiv \sum_w \alpha_{kw}$ and $c_j \equiv \sum_w c_{jw}$.

We update cluster assignment probabilities and hyperparameters of Dirichlet priors with the EM algorithm described below.

E step: Update p_{jk} by

$$p_{jk} \leftarrow \frac{\sum_j p_{jk}}{\sum_j \sum_k p_{jk}} \cdot \frac{\Gamma(\alpha_k)}{\Gamma(c_j + \alpha_k)} \prod_w \frac{\Gamma(c_{jw} + \alpha_{kw})}{\Gamma(\alpha_{kw})}$$

and then normalize p_{jk} by $p_{jk} \leftarrow p_{jk} / \sum_k p_{jk}$.

M step: Update α_{kw} by

$$\alpha_{kw} \leftarrow \alpha_{kw} \cdot \frac{\sum_j p_{jk} \{\Psi(c_{jw} + \alpha_{kw}) - \Psi(\alpha_{kw})\}}{\sum_j p_{jk} \{\Psi(c_j + \alpha_k) - \Psi(\alpha_k)\}}$$

where $\Gamma(\cdot)$ is gamma function, and $\Psi(\cdot)$ is digamma function. The update formula for α_{kw} is based on Minka's discussion (Minka, 2000). We terminate the iteration of E and M steps when the log likelihood increases by less than 0.001%.

Before entering into the loop of E and M steps, we initialize α_{jk} to 1, because this makes Dirichlet priors uniform distributions. Further, we initialize p_{jk} not randomly but by the EM algorithm for multinomial mixtures (Nigam et al., 2000). In the EM for multinomial mixtures, we use a random initialization for p_{jk}. The execution of EM for multinomial mixtures is repeated 30 times each from a random initialization. Each of the 30 executions of the EM for multinomial mixtures gives a different estimation of p_{jk}. Therefore, we choose the estimation giving the largest likelihood as the initial setting of p_{jk} in the EM algorithm for DCM. We conduct this entire procedure three times. Then, among the three clustering results, we select the result giving the largest likelihood as the final output of our document clustering.

The time complexity of our EM algorithm is $O(IKM)$, where I is the number of iterations and M is the number of different pairs of document and word. In general, M is far smaller than $J \times W$ due to the sparseness discussed above.

4 EVALUATION EXPERIMENT

4.1 Procedure

The following two document sets were used in our evaluation experiment:

- The one is the set of Korean newswire articles downloaded from the Web site of *Seoul Newspaper*[1]. We denote this dataset as SEOUL. This set consists of 52,730 articles whose dates range from January 1st, 2008 to September 30th, 2009. Each article belongs to one among the following four categories: *Economy*, *Local issues*, *Politics*, and *Sports*. Therefore, we set $K = 4$ in DCM.
- The other is the set of Chinese newswire articles dowloaded from *Xinhua Net*[2]. We denote this dataset as XINHUA. This set consists of 20,127 articles whose dates range from May 8th to December 17th in 2009. All articles are written in simplified Chinese. Each article belongs to one among the three categories: *Economy*, *International*, and *Politics*. Therefore, we set $K = 3$.

We regarded article categories as the ground truth when we evaluated document clusters.

To compare with maximal substrings, we extracted words by applying KLT morphological analyzer (Gang, 2009) to Korean articles. On the other hand, we applied a word segmenter, implemented by using L1-regularized linear conditional random fields (CRF) (Sutton and McCallum, 2007), to Chinese articles. Our algorithm for parameter optimization in training this Chinese word segmenter is based on stochastic gradient descent algorithm with exponential decay scheduling (Tsuruoka et al., 2009). This segmenter achieved the following F-scores for the four datasets of SIGHAN Bakeoff 2005 (Tseng et al., 2005): 0.943 (AS), 0.941 (HK), 0.929 (PK) and 0.960 (MSR). In our experiment, we used the segmenter trained with MSR dataset, because this dataset gave the highest F-score. For Korean language, we could not find any training data comparable with the SIGHAN training data in its size and quality. Therefore, we used KLT for Korean word segmentation.

The wall clock time required for extracting all maximal substrings was only a few minutes for both datasets on a PC equipped with Intel Core 2 Quad 9650 CPU. This wall clock time is comparable with the time required for word extraction by our Chinese segmenter, though the time required for training the segmenter is not included. Further, it is much less than the time required for Korean morphological analysis, because KLT achieves its excellence by dictionary lookups. While KLT can provide part-of-speech tags, they are not required for our experiment.

The running time of document clustering is proportional to M, i.e., the number of different pairs of document and word (or of document and maximal substring). When we extracted words by KLT morphological analyzer from SEOUL dataset, M was equal to 8,208,591 after removing the words whose frequencies are less than five. When we used maximal substrings, M was 37,079,130 after removing

[1] http://www.seoul.co.kr/

[2] http://www.xinhuanet.com/

the maximal substrings whose frequencies are less than five. Further, when we extracted words by the CRF-based segmenter from XINHUA dataset, M was 3,244,859 after removing the words whose frequencies are less than five. When we used maximal substrings, M was 17,561,135 after removing the maximal substrings whose frequencies are less than five.

We evaluated the quality of clusters as follows:

1. We calculated precision and recall for each cluster. Precision is defined as

$$\frac{\#(\text{true positive})}{\#(\text{true positive}) + \#(\text{false positive})},$$

and recall is defined as

$$\frac{\#(\text{true positive})}{\#(\text{true positive}) + \#(\text{false negative})},$$

where "#" means the number;

2. We calculated *F-score* as the harmonic mean of precision and recall for each cluster; and
3. The F-score was averaged over all clusters.

We used the resulting averaged F-score as our evaluation measure.

We executed document clustering 64 times for each setting of n_L and n_H. Consequently, we obtained 64 F-scores for each setting. The effectiveness of each setting of n_L and n_H was represented by the mean and the standard deviation of these 64 F-scores.

We reduced the variety of maximal substrings as follows. We removed the maximal substrings whose frequency was less than n_L. We tested the following six settings for n_L: 10, 20, 50, 100, 200, and 500. We present the evaluation result for each setting of n_L in Table 1, where the mean and the standard deviation are computed over the 64 F-scores for each case.

With respect to the best setting of n_L, we tested the following five settings for c_H to specify n_H: 0.2, 0.1, 0.05, 0.02, and 0.01. For example, when c_H was set to 0.02, n_H was set to $0.02n_1$. The result for each setting of c_H is given in Table 2 also with the mean and the corresponding standard deviation.

The same reduction method was applied not only to maximal substrings but also to words extracted by the morphological analyzer from SEOUL dataset and to words extracted by our segmenter from XINHUA dataset. The evaluation results for these *supervised* word extraction methods are also given in Table 1 and Table 2.

4.2 Analysis of Results

We can analyze the results presented in Table 1 and Table 2 as follows.

In Table 1, the top panel gives the results for SEOUL dataset, and the bottom panel for XINHUA dataset. The column labeled as "n_L" includes the settings for n_L, and the column labeled as "# words" includes the numbers of words or the numbers of maximal substrings after removing low frequency features. We regard the boldfaced cases as the best setting for n_L, because each of these cases leads to the F-score larger than the other settings. We can obtain the following observations from Table 1:

- For SEOUL dataset, the F-scores achieved with maximal substrings were *quite close* to those achieved with words extracted by the morphological analyzer. It can be said that the results prove the effectiveness of maximal substrings.
- For XINHUA dataset, maximal substrings gave weaker results than words obtained by the CRF-based segmenter. Further, bag of maximal substrings representation led to larger standard deviations of F-scores. These large standard deviations suggest that there is a room for improvement regarding with selection of maximal substrings or with clustering method so as to utilize maximal substrings more effectively.
- By removing more low frequency words or maximal substrings, a smaller standard deviation was achieved. A large standard deviation means that the quality of document clusters is quite different trial by trial. Therefore, it is desirable to reduce as many low frequency words as possible. However, the reduction of too many low frequency words resulted in a large drop of F-scores. Table 1 shows that the range $n_L \leq 200$ is recommended.
- We may expect that low frequency words will be sharply related to a specific topic and thus will have a discriminative power. However, in our case, by using more low frequency words, we obtained larger standard deviations. This may tell that most low frequency words misled clustering process maybe by occasionally focusing on a minor aspect of the document content.

We further check if we can achieve an improvement by reducing also high frequency features. Especially for XINHUA dataset, we could not obtain any good F-scores only with the reduction of low frequency maximal substrings. Therefore, we next discuss about the results presented in Table 2.

In Table 2, the top panel gives the results for SEOUL dataset, and the bottom panel for XINHUA dataset. Each of the boldfaced cases corresponds to the best F-score among the various settings of c_H. For ease of comparison, Table 2 also presents the best cases from Table 1 in the rows that have the value 1.0

Table 1: Evaluation of clusters obtained after removing only low frequency features.

SEOUL (52,730 docs, 4 clusters)			
	n_L	# words	F-score
Maximal Substrings	10	186,032	0.826±0.049
	20	126,619	0.854±0.036
	50	72,104	0.867±0.029
	100	**45,360**	**0.872±0.004**
	200	26,923	0.869±0.002
	500	12,590	0.856±0.001
Morphological Analysis	10	61,416	0.859±0.043
	20	37,800	0.883±0.025
	50	**20,068**	**0.892±0.002**
	100	12,411	0.890±0.001
	200	7,620	0.886±0.000
	500	3,798	0.879±0.002

XINHUA (20,127 docs, 3 clusters)			
	n_L	# words	F-score
Maximal Substrings	10	285,438	0.650±0.051
	20	**140,690**	**0.690±0.051**
	50	53,239	0.672±0.039
	100	25,344	0.649±0.014
	200	12,351	0.642±0.002
	500	5,049	0.619±0.002
Word Segmentation	10	23,234	0.753±0.021
	20	15,347	0.750±0.021
	50	8,783	0.755±0.012
	100	5,709	0.760±0.007
	200	**3,596**	**0.762±0.007**
	500	1,752	0.741±0.011

Table 2: Evaluation of clusters obtained after removing both high and low frequency features.

SEOUL (52,730 docs, 4 clusters)			
	c_H	# words	F-score
Maximal Substrings ($n_L = 100$)	1.0	45,360	0.872±0.004
	0.2	45,320	0.873±0.008
	0.1	**45,260**	**0.875±0.007**
	0.05	45,159	0.875±0.008
	0.02	44,918	0.873±0.011
	0.01	44,589	0.874±0.011
Morphological Analysis ($n_L = 50$)	**1.0**	**20,068**	**0.892±0.002**
	0.2	20,056	0.890±0.007
	0.1	20,040	0.890±0.008
	0.05	19,975	0.888±0.009
	0.02	19,721	0.887±0.009
	0.01	19,179	0.887±0.009

XINHUA (20,127 docs, 3 clusters)			
	c_H	# words	F-score
Maximal Substrings ($n_L = 20$)	1.0	140,690	0.690±0.051
	0.2	**140,672**	**0.701±0.059**
	0.1	140,611	0.693±0.056
	0.05	140,466	0.696±0.051
	0.02	140,069	0.685±0.051
	0.01	139,515	0.698±0.047
Word Segmentation ($n_L = 200$)	1.0	3,596	0.762±0.007
	0.2	**3,593**	**0.762±0.001**
	0.1	**3,587**	**0.762±0.001**
	0.05	**3,559**	**0.762±0.001**
	0.02	3,468	0.718±0.000
	0.01	3,261	0.754±0.001

at the column labeled as c_H. The case $c_H = 1.0$ corresponds to the case where we reduce no high frequency features, i.e., the case considered in Table 1.

Table 2 provides the following observations:

- For bag of words representation, the reduction of high frequency features did not lead to any improvement on both SEOUL dataset and XINHUA dataset. The reduction of high frequency features just worked as a reduction of working space for document clustering.
- For bag of maximal substrings representation, the reduction of high frequency features led to a small improvement. However, the improvement was not statistically significant. Therefore, also for maximal substrings, the reduction only worked as a reduction of working space. Of course, the same observation can be restated as follows: we could reduce the number of elementary features without harming cluster quality.
- For XINHUA dataset, even after the reduction of high frequency features, bag of maximal substrings representation provided weaker results than bag of words representation. This means that we should use a supervised segmenter for Chinese documents as long as we can prepare a sufficient amount of training data.

We can conclude that maximal substrings are as effective as words extracted by the dictionary-based morphological analysis for Korean documents. This may be partly because Korean sentences contain white spaces and thus maximal substring extraction works as a fine improvement of this intrinsic segmentation. In contrast, for Chinese documents, we need some more sophistication to make maximal substrings equally effective.

However, we think that the difference of effectiveness between bag of words representation and bag of maximal substrings representation is not so large to make the latter inapplicable in document clustering.

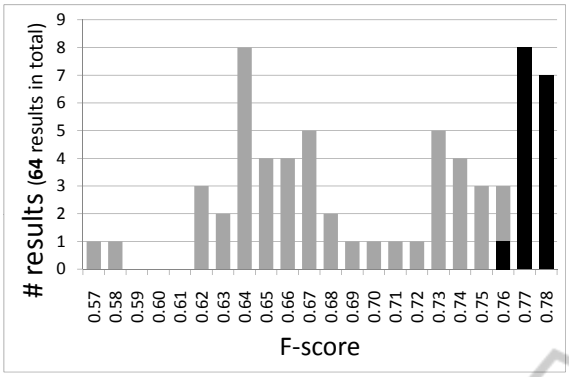

Figure 2: Distribution of F-scores achieved with maximal substrings for XINHUA dataset. This histogram shows the 64 F-scores obtained by setting $n_L = 20$ and $c_H = 0.2$. F-scores are rounded off to two decimal places. While the mean of all 64 F-scores is 0.701, we have 16 F-scores (black part of the histogram) that are larger than the best mean F-score 0.762 achieved with CRF-based word segmentation (cf. the bottom panel in Table 2).

Even with respect to XINHUA dataset, the cluster quality obtained with bag of maximal substrings representation is acceptable, as we discuss below.

When we set $n_L = 200$, $c_H = 0.2$ and reduce the variety of words extracted by our CRF-based segmenter from XINHUA dataset, the best mean F-score 0.762 was achieved (cf. Table 2). However, when we set $n_L = 20$ and $c_H = 0.2$ and reduce the variety of maximal substrings, we could obtain F-scores larger than 0.762 for 16 clustering results among all 64 clustering results. That is, 25% of the clustering results showed a quality better than the mean quality of document clusters bag of words representation gave.

Figure 2 presents the detailed distribution of the F-scores achieved with bag of maximal substrings representation when we set $n_L = 20$ and $c_H = 0.2$ for XINHUA dataset. In this histogram, F-scores are rounded off to two decimal places. The black part of the histogram corresponds to the 16 F-scores that were larger than the best mean F-score 0.762 achieved with bag of words representation.

In addition, for the case where we achieved the best mean F-score 0.762 with bag of words representation, all 64 F-scores fell within the interval $[0.755, 0.765)$. Further, the best three F-scores were 0.764, 0.764, and 0.763. In contrast, the best three F-scores were 0.782, 0.781, and 0.780 when we used maximal substrings as elementary features and set $n_L = 20$, $c_H = 0.2$.

Therefore, it is a promising work to introduce some sophistication into selection of maximal substrings and also into document clustering so as not to occasionally give extremely poor clustering results.

5 CONCLUSIONS

As text data originating from SNS environments come to show a wider divergence in the writing style or in the used vocabularies, unsupervised extraction of elementary features becomes more important as a preprocessing for various text mining techniques than before. Therefore, in this paper, we provide experimental results where we compare bag of maximal substrings representation with bag of words representation, because maximal substrings can be extracted in a totally unsupervised manner.

Our results showed that maximal substrings were not equally effective with words extracted by the elaborated supervised method for Chinese documents, though we could obtain impressive results for Korean documents. We need a more sophisticated selection method to obtain a special subset of maximal substrings for Chinese documents. We should also improve document model for clustering to capture frequency statistics of maximal substrings more cleverly than DCM. We think that these future works are worthy to do based on the reason discussed in the late part of the previous section.

Further, if we use larger datasets, we can expect that more reliable statistics of maximal substrings will be obtained and thus that more convincing evaluation results will be provided. It is an important future work to acquire a more realistic insight with respect to the trade-off between the following two types of cost:

- the cost to improve the effectiveness of maximal substrings with a more elaborated selection method and/or with a more elaborated clustering method; and
- the cost to prepare training data for supervised feature extraction and then to train a segmentation model by using the data.

We also have a plan to conduct experiments where we use maximal substrings as elementary features for a multi-topic analysis based on latent Dirichlet allocation (Blei et al., 2003) for text data or for DNA/RNA sequence data (Chen et al., 2010).

ACKNOWLEDGEMENTS

This work was supported in part by Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Young Scientists (B) 60413928 and also by Nagasaki University Strategy for Fostering Young Scientists with funding provided by Special Coordination Funds for Promoting Science and Technology of the

Ministry of Education, Culture, Sports, Science and Technology (MEXT).

REFERENCES

Abouelhoda, M., Ohlebusch, E., and Kurtz, S. (2002). Optimal exact string matching based on suffix arrays. In *SPIRE'02, the Ninth International Symposium on String Processing and Information Retrieval*, pages 31–43.

Blei, D., Ng, A., and Jordan, M. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Chen, X., Hu, X., Shen, X., and Rosen, G. (2010). Probabilistic topic modeling for genomic data interpretation. In *BIBM'10, IEEE International Conference on Bioinformatics & Biomedicine*, pages 18–21.

Choi, K., Isahara, H., Kanzaki, K., Kim, H., Pak, S., and Sun, M. (2009). Word segmentation standard in Chinese, Japanese and Korean. In *the 7th Workshop on Asian Language Resources*, pages 179–186.

Chumwatana, T., Wong, K., and Xie, H. (2010). A SOM-based document clustering using frequent max substrings for non-segmented texts. *Journal of Intelligent Learning Systems & Applications*, 2:117–125.

Gang, S. (2009). Korean morphological analyzer KLT version 2.10b. http://nlp.kookmin.ac.kr/HAM/kor/.

Kasai, T., Lee, G., Arimura, H., Arikawa, S., and Park, K. (2001). Linear-time longest-common-prefix computation in suffix arrays and its applications. In *CPM'01, the 12th Annual Symposium on Combinatorial Pattern Matching*, pages 181–192.

Li, Y., Chung, S., and Holt, J. (2008). Text document clustering based on frequent word meaning sequences. *Data & Knowledge Engineering*, 64:381–404.

Madsen, R., Kauchak, D., and Elkan, C. (2005). Modeling word burstiness using the Dirichlet distribution. In *ICML'05, the 22nd International Conference on Machine Learning*, pages 545–552.

Minka, T. (2000). Estimating a Dirichlet distribution. http://research.microsoft.com/en-us/um/people/minka/papers/dirichlet/.

Mochihashi, D., Yamada, T., and Ueda, N. (2009). Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling. In *ACL/IJCNLP'09, Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the Fourth International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 100–108.

Navarro, G. and Makinen, V. (2007). Compressed full-text indexes. *ACM Computing Surveys (CSUR)*, 39(1).

Nigam, K., McCallum, A., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134.

Nong, G., Zhang, S., and Chan, W. (2008). Two efficient algorithms for linear time suffix array construction. http://doi.ieeecomputersociety.org/10.1109/TC.2010.188.

Okanohara, D. and Tsujii, J. (2009). Text categorization with all substring features. In *SDM'09, 2009 SIAM International Conference on Data Mining*, pages 838–846.

Poon, H., Cherry, C., and Toutanova, K. (2009). Unsupervised morphological segmentation with log-linear models. In *NAACL/HLT'09, North American Chapter of the Association for Computational Linguistics - Human Language Technologies 2009 Conference*, pages 209–217.

Sutton, C. and McCallum, A. (2007). An introduction to conditional random fields for relational learning. In *Introduction to Statistical Relational Learning*, pages 93–128.

Teh, Y. (2006). A hierarchical Bayesian language model based on Pitman-Yor processes. In *COLING/ACL'06, Joint Conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics*, pages 985–992.

Tseng, H., Chang, P., Andrew, G., Jurafsky, D., and Manning, C. (2005). A conditional random field word segmenter for SIGHAN bakeoff 2005. In *the Fourth SIGHAN Workshop*, pages 168–171.

Tsuruoka, Y., Tsujii, J., and Ananiadou, S. (2009). Stochastic gradient descent training for L1-regularized log-linear models with cumulative penalty. In *ACL/IJCNLP'09, Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the fourth International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, pages 477–485.

Wang, X. and McCallum, A. (2006). Topics over time: a non-Markov continuous-time model of topical trends. In *KDD'06, the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 424–433.

Zhang, D. and Dong, Y. (2004). Semantic, hierarchical, online clustering of Web search results. In *APWeb'04, the Sixth Asia Pacific Web Conference*, pages 69–78.

Zhang, D. and Lee, W. (2006). Extracting key-substring-group features for text classification. In *KDD'06, the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 474–483.