2014 IEEE International Conference on Data Mining Workshop

Domain-independent Unsupervised Text Segmentation For Data Management

Makoto Sakahara
Tokyo Institute of Technology
Japan
sakahara@ntt.dis.titech.ac.jp

Shogo Okada
Tokyo Institute of Technology
Japan
okada@dis.titech.ac.jp

Katsumi Nitta
Tokyo Institute of Technology
Japan
nitta@dis.titech.ac.jp

***Abstract*—In this study, we have proposed a domain-independent unsupervised text segmentation method, which is applicable to even if unseen single document. This proposed method segments text documents by evaluating similarity between sentences. It is generally difficult to calculate semantic similarity between words that comprise sentences when the domain knowledge is insufficient. This problem influences segmentation accuracy. To address this problem, we use word2vec to calculate semantic similarity between words. Using word2vec, we embed semantic relationships between words in a vector space by training with large domain-independent corpora. Furthermore, we combine semantic and collocation similarities, i.e., the features between words within a document. The proposed method applies this combined similarity to affinity propagation clustering. Similarity between sentences is defined based on the earth mover's distance between the frequencies of the obtained topical clusters. After calculating similarity between sentences, segmentation boundaries are automatically optimized using dynamic programming. The experimental results obtained using two datasets show that the proposed method clearly outperforms state-of-the-art domain-independent approaches and obtains equal performance with state-of-the-art domain-dependent approaches such as those that use topic modeling.**

I. Introduction

Meta information, which represents the contents of data, is useful for the marketing of data. Meta information may be an abstract of the data, a genre which the data belongs to, the bibliographic information, or reputation. This information is used to recommend or retrieval data which match the consumer's needs. If we extract detail meta information from the data, the performance of recommendation and retrieval is improved. However, burden of extracting meta information is generally very heavy.

When we treat text documents or movie data transcripts, if we segment the data into several topics or scenes, the extraction of meta information becomes easier. In information retrieval, if topic boundaries are provided, the user can focus on an effective range that is relevant to their query without considering the entire document. This is useful for other natural language processing tasks such as information recommendation or summarization, and is applicable to data marketing management. Text segmentation aims to divide a document into topical segments. In addition to boundary information, if each segment is representative of a topic, such information is more useful for subsequent tasks.

TextTiling[1] is a classic and simple unsupervised text segmentation method, whose method calculates the cosine distance between two adjacent blocks. Each block is represented by a term frequency vector. This method can be applied to any text in any domain, even if training data are not available. Rather than using the raw term frequency vector, some approaches, e.g., BAYESSEG[2], [3], TopicTiling[4], TSM[5], NTSeg[6], employ topic modeling to improve performance, such as latent Dirichlet allocation[7]. Earlier topic modeling-based approaches [2], [3], [4] typically capture the document-level word collation implicitly to reveal topics, which can result in mismatched semantics between the obtained topics, and word, sentence, and paragraph levels in the document. In addition such models assume that there is no correlation between topics. Therefore, these approaches cannot deal with collation between topics; e.g., a topic about economics is also more likely to be about finance than science. To address those problems, some approaches [5], [6] employ segment, word levels topic modeling to improve performance. These topic modeling-based approaches require domain corpus for training data. It has been previously reported that domain unadapted trained model decreases performance significantly, even if employing domain similar corpus for training, decreases slightly by [3]. Subsequent tasks occasionally deal with domain-independent content; therefore, when applying real data, this requirement can introduce problems.

Therefore, we introduce a method that measures similarity between each word for clustering and each sentence for segmentation, which is applicable to even if unseen single document.

To measure similarity between words, we combine two different similarities, i.e., semantic similarity obtained from word2vec [8], [9] and collocation similarity obtained from within the document. word2vec computes the distributed representations of words from huge datasets. The semantics and relationships between words are embedded in the vector space through model training. When the model is well trained, it is possible to identify similar words in terms of semantics to measure the cosine similarity between words. Collocation similarity means how similar frequency of words appearance around those.

We apply this combined similarity to affinity propagation[10] to obtain topical clusters, that are mapped to each word. We assume that each sentence consists of more than one topical cluster represented by words.

After word clustering is performed, each sentence is represented as a frequency histogram of word clusters. We then

978-1-4799-4274-9/14 $31.00 © 2014 IEEE
DOI 10.1109/ICDMW.2014.118

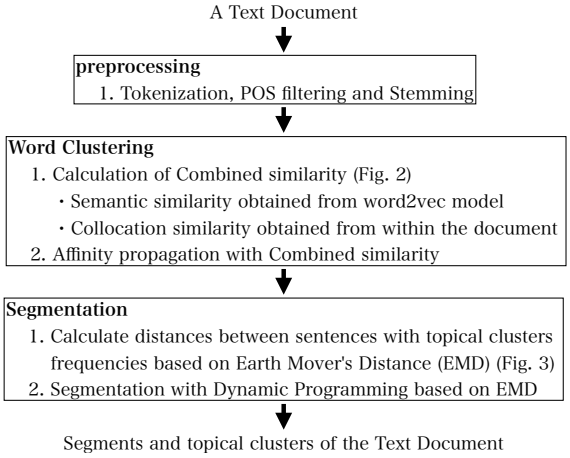

Fig. 1. Basic concept of proposed.

calculate the similarity between each sentence on the basis of the earth mover's distance (EMD) with the frequencies of topical clusters. EMD is a metric between two distributions that considers the distance between each cluster and their weights. Each segment is represented by adjacent sentences. Note that text segmentation requires the detection of boundaries for each sentence. It has been previously reported that text segmentation that uses dynamic programming (DP) can improve boundary detection accuracy [3], [11], [12]. Sliding a window to detect boundaries, which is employed in TextTiling[1], can obtain local optimization; however, DP techniques provide globally optimized segmentation. Therefore, we employ DP techniques to optimize segments based on the EMD of each sentence.

We provide two main contributions. First, we propose a topical clustering method based on combined similarity for text segmentation rather than topic model approaches. Second, we define the distance between sentences based on EMD with the frequencies of topical clusters to consider the correlation of those clusters.

The remainder of this paper is organized as follows. In Section 2, we introduce the proposed method. In Section 3, we compare the proposed method with state-of-the-art domain-independent and dependent approaches. Conclusions and suggestions for future work are given in Section 4.

II. Proposed Method

The proposed method comprises three steps. First, preprocessing, which consists of tokenization, stemming, and part of speech (POS) filtering. Second, calculation of the combined similarity between each word consists of obtaining the semantic similarity using word2vec and collocation similarity obtained from within the document. The combined results are then applied to clustering. Third, we calculate the distances between sentences based on EMD with the frequencies of the obtained topical clusters considering the correlation of those clusters and the optimization of segments based on the distances obtained via DP. The basic concept of the proposed method is shown in Fig .1.

A. Preprocessing

First, a document is tokenized into a word stream. Second, the word stream is filtered by POS. Note that we only use nouns, proper nouns, verbs, and adjectives, and that the Porter stemming method[13] is applied. Words in a document are denoted $(w_1, w_2, \ldots, w_i, \ldots, w_z)$. This is the word stream. Therefore, for example, w_1 may or may not be the same as w_2.

B. Word Clustering

In domain-independent text segmentation, a lack of background knowledge between each word obtained from within a single document is a problem that leads to poor clustering results. To address this problem, we use word2vec [8], [9]. Each word in a document is input to the word2vec model, and the cosine similarity between words is measured. The obtained matrix is referred to as the semantic similarity matrix S_{sem}. The projected word set is represented by $\{w_1, w_2, w_3, \ldots, w_i, \ldots, w_n\}$. Note that all words in this set are unique.

Furthermore, we calculate the collocation similarity matrix S_{col}. We count the frequencies of words appearing near the central word of a window via sliding. For a window size of 2, w_i's initial collocation vector is represented by $\{w_{i-2}, w_{i-1}, w_{i+1}, w_{i+2}\}$. Available words are counted when the window reaches the edge of the word stream. Therefore, w_1's initial collocation vector is represented by $\{w_2 : 1, w_3 : 1\}$. If the same word appears in the word stream, a collocation vector is added. To measure the cosine similarity between collocation vectors and obtaining collocation similarity matrix S_{col}. Note that the semantic similarity and collocation similarity matrices (S_{sem} and S_{col}, respectively) have equal dimensionality n. We then combine the semantic and collocation similarity matrices as follows.

$$S_{com} = \alpha \frac{(1 + S_{sem})}{2} + (1 - \alpha) S_{col} \tag{1}$$

$$S_{sem} = \begin{pmatrix} 1.0 & sim_s(w_1, w_2) & \ldots & sim_s(w_1, w_n) \\ sim_s(w_2, w_1) & 1.0 & \ldots & sim_s(w_2, w_n) \\ \vdots & \vdots & \ddots & \vdots \\ sim_s(w_n, w_1) & sim_s(w_n, w_2) & \ldots & 1.0 \end{pmatrix}$$

$$S_{col} = \begin{pmatrix} 1.0 & sim_c(w_1, w_2) & \ldots & sim_c(w_1, w_n) \\ sim_c(w_2, w_1) & 1.0 & \ldots & sim_c(w_2, w_n) \\ \vdots & \vdots & \ddots & \vdots \\ sim_c(w_n, w_1) & sim_c(w_n, w_2) & \ldots & 1.0 \end{pmatrix}$$

Here, α is the mixture ratio. S_{sem} is scaled between zero and one. We show the concept of the combined similarity in Fig. 2. We use matrix S_{com} as an input for word clustering.

1) Clustering Algorithm: Note that $S_{combined}$ does not satisfy triangle inequality. Therefore, the clustering algorithm in the proposed method needs to allow unsatisfied triangle inequality. Moreover, in text segmentation, the number of topical clusters can not be predefined.

Thus, we employ affinity propagation[10], which takes similarity measures between pairs of data points as input. Then, real-valued messages are high-quality set of exemplars and corresponding clusters gradually emerges. Affinity propagation can handle similarity that is not symmetric or does not satisfy triangle inequality and obtain the number of clusters automatically. Moreover, this does not depend on random initialization, such as k-means. Thus, this clustering algorithm is suitable for our intended purpose.

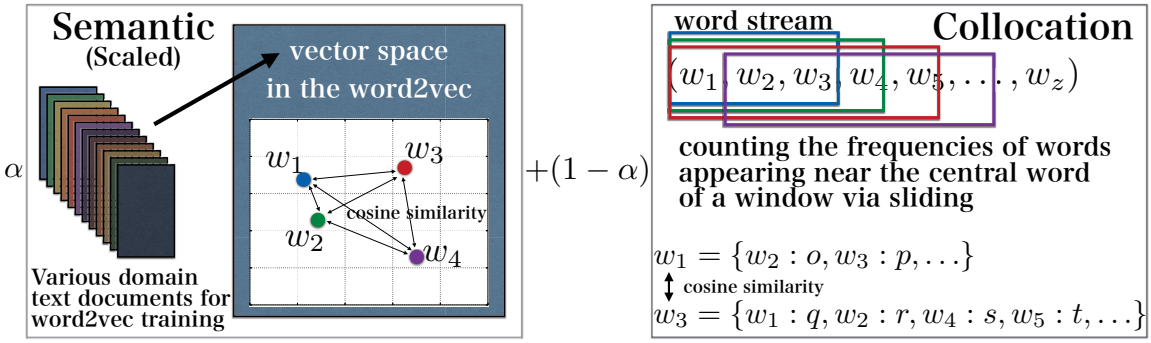

Fig. 2. Combined semantic and collocation similarity (Eq. (1)). α is mixture ratio.o, p, q, r, s, t are frequencies of each word. Various domain text documents (e.g., Wikipedia) are separated from target domain.

C. Segmentation

1) Distances between sentences with topical clusters frequencies based on EMD: We obtain topical clusters about a document via word clustering. Next, the proposed method measures similarity between each sentence based on EMD [14] with the frequencies of topical clusters to capture the correlation of those clusters.

The EMD is a metric between two distributions defined as the minimum amount of work required to change one signature into another. The notion of work is based on the user-defined ground distance which is the distance between two features. Computation of EMD is based on a solution to the transportation problem, which can be formalized as follows. Let $P = \{(p_1, w_{p_1}), \ldots, (p_m, w_{p_m})\}$ be the first signature with m clusters, where p_i is the cluster representative and w_{p_i} is the weight of the cluster. $Q = \{(q_1, w_{q_1}), \ldots, (q_n, w_{q_n})\}$ is the second signature with n clusters. Matrix D is the ground distance matrix, where d_{ij} is the distance between clusters p_i and q_j. Matrix F is the flow matrix, where f_{ij} is the flow between p_i and q_j that minimizes the overall cost.

$$WORK(P,Q,F) = \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} d_{ij} \tag{2}$$

Eq. (2) is subject to the following constraints:

$$f_{ij} \geq 0 \qquad 1 \leq i \leq m,\ 1 \leq j \leq n \tag{3}$$
$$\sum_{j=1}^{n} f_{ij} \leq w_{p_i} \qquad 1 \leq i \leq m \tag{4}$$
$$\sum_{i=1}^{m} f_{ij} \leq w_{q_j} \qquad 1 \leq j \leq n \tag{5}$$
$$\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = min\left(\sum_{i=1}^{m} w_{p_i}, \sum_{j=1}^{n} w_{q_j}\right) \tag{6}$$

Note that Constraint 3 allows moving supplies from P to Q. Constraint 4 limits the amount of supplies that can be sent by clusters in P to their weights. Constraint 5 limits the amount of supplies that can be received by clusters in Q to their weights. Constraint 6 forces the movement of the maximum amount of supplies. Once the transportation problem is solved, and the optimal flow F is determined, the EMD is defined as the resulting work normalized by total flow as follows.

$$EMD(P,Q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} d_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}} \tag{7}$$

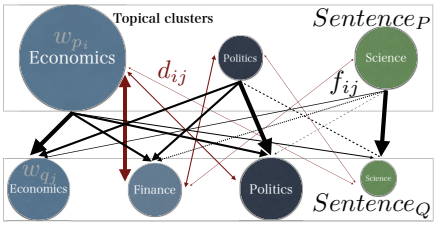

Fig. 3. Calculating distance between sentences based on EMD with frequencies of obtained topical clusters and their correlations. w_{p_i} and w_{q_j} are weight of the cluster (i.e., frequency of the cluster in sentence; represented by circle size). Larger circles indicates greater frequency in the sentence. d_{ij} is the distance between each topical cluster (red bidirectional arrow). Larger arrows and similar color of circles indicate cluster similarity. f_{ij} is the flow of each topical cluster(black arrow). Larger solid line arrows indicates greater flow.

We define the distance between sentences on the basis of EMD. An example of this calculation is shown in Fig. 3. Here a document consists of $(1,2,\ldots,P,Q,\ldots,N)$ sentences. p_i and q_j are topical clusters, $sentence_{P_{w_{p_i}}}$ and $sentence_{Q_{w_{q_j}}}$ are the frequencies of topical clusters in each sentence. Thus, $\sum_{i=1}^{m} sentence_{P_{w_{p_i}}} = 1$ and $\sum_{j=1}^{n} sentence_{Q_{w_{q_j}}} = 1$; therefore, Constraints 4 and 5 are satisfied. The ground distance matrix D consists of the distance between topical clusters p_i and q_j as follows.

$$d_{ij} = \frac{1}{|p_i||q_j|}\sum_{w_1 \in p_i}\sum_{w_2 \in q_j} d(w_1, w_2) \tag{8}$$

Here, $d(w_1,w_2) = 1 - sim_{combined}(w_1,w_2)$ This is commonly called the group average method. If EMD is a low value, $sentence_P$ is topically similar to $sentenc_Q$.

2) Segmentation with DP based on EMD: Text segmentation can be implemented efficiently with DP techniques [3], [11], [12]. Note that the combinatorial argument is 2^{n-1}

segmentations for n sentences. DP techniques can reduce computational cost. Text segmentation is explained by a directed acyclic graph using DP. Scores for all the possible node pairs are computed using DP. Therefore, if the graph contains n nodes, $n \times (n+1)/2$ node pairs must be considered. Before explaining the proposed DP approach, we introduce the base DP method proposed by [12].

Fragkou DP: Suppose a document contains N sentences and has a vocabulary of W distinct words. Consider the $N \times W$ matrix A defined as follows:

$$A_{n,w} = \begin{cases} 1 & \text{if the } w\text{-th word appears in the } n\text{-th sentence,} \\ 0 & \text{else.} \end{cases} \tag{9}$$

Hence, $n = 1,2,\ldots,N$, $w = 1,2,\ldots,W$, and A is a sentences$\times$ words matrix. Fragkou et al. define the $A \times A$ similarity matrix D between sentences of the document as follows:

$$D_{m,n} = \begin{cases} 1 & \text{if } \sum_{w=1}^{W} A_{m,w} A_{n,w} > 0, \\ 0 & \text{if } \sum_{w=1}^{W} A_{m,w} A_{n,w} = 0. \end{cases} \tag{10}$$

Here, $m,n = 1,2,\ldots,N$. Therefore, $D_{m,n} = 1$ if the m-th and t-th sentences have at least one common word. Good segmentation should maximize the density of 1's in the submatrices of D, which correspond to segments. Their considerations can be formalized by defining the segmentation cost function J as follows.

$$J = \sum_{k=1}^{K} \Bigg(\alpha \cdot G\Big(\frac{n_k - n_{k-1} - \mu}{\sigma}\Big) - (1-\alpha) \cdot \frac{\sum_{m=n_{k-1}+1}^{n_k} \sum_{n=n_{k-1}+1}^{n_k} D_{m,n}}{(n_k - n_{k-1})^r} \Bigg) \tag{11}$$

The total segmentation cost J is the sum of the costs of the K segments, and the cost of each segment is the sum of two terms. $G(\frac{n_k - n_{k-1} - \mu}{\sigma})$ is a length cost function obtained by prior knowledge about the average segment length μ. The right term is the generalized density of a segment.

Proposed DP: Here, we extend the composition of similarity matrix D between sentences. We compose connectivity costs matrix C from EMD, which is predefined rather than using common word existence (i.e., Eq. (10)) as follows.

$$C_{P,Q} = \begin{cases} 0 & \text{if } P = Q, \\ 1 - EMD(P,Q) & \text{if } P,Q \notin \emptyset, \\ mean(C_{P_{-1},Q}, C_{P_{+1},Q}) & \text{if } P \in \emptyset. \end{cases} \tag{12}$$

Here, C is symmetric matrix. EMD is distance; therefore $1 - EMD$ is the connectivity cost between sentences. A short sentence usually contains no term after preprocessing. If there are no terms in the sentence, calculate the mean around sentence for smoothing. This is based on the assumption that proximal sentence are similar relative to the topic.

Unlike artificial datasets, it is difficult to obtain prior knowledge from real datasets about segments such as the number of segments and their average size. Therefore, we redefine the following segmentation cost function $\hat{J}$ as follows.

$$\hat{J} = \sum_{k=1}^{K} \frac{\sum_{P=n_{k-1}+1}^{n_k} \sum_{Q=n_{k-1}+1}^{n_k} C_{P,Q}}{n_k - n_{k-1}} \tag{13}$$

This is the performed by removing the left term (i.e., the length cost function) from Eq. (11), replacing S with C and setting $r = 1$. We maximize this cost through DP techniques.

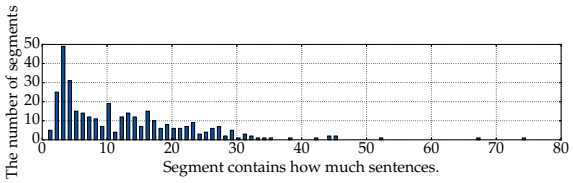

Fig. 4. Each segment contains how much sentences on transcripts of televised news programs dataset.

III. Experiments

A. Datasets

We evaluated the proposed method using two datasets. The first is a dataset in which each document is a chapter selected from a medical textbook [2]. This dataset is often used as a benchmark in text segmentation. Here, the task is to divide each chapter into the sections indicated by the author. This dataset contains 227 chapters and 1136 sections.

For the second dataset, we prepared transcripts from one month of televised news programs (NHK News 7, an early evening Japanese news program). The reference boundary indicates a topic change in a program, which was annotated manually. This dataset contains 31 programs and 318 boundaries. We show each segment contains how much sentences in Fig 4. Segments consisting of few sentences are most frequent, and most segments contains less than 30 sentences.

B. Metrics

In text segmentation, precision and recall are insufficient. These penalizes whether the predicted boundary is near or far from a reference boundary. To overcome this problem, P_k[15] and $WindowDiff$[16] are employed, which are widely used metrics in text segmentation. The value of k is half the average reference segment size. These metrics compute penalty via sliding window size of k. Note that lower value indicates better performance for each metric. It is described that $1-WindowDiff$ is the effective accuracy of text segmentation by [17].

C. Settings

The proposed method requires a trained word2vec model. The word2vec model is trained using all current revisions of articles from Wikipedia as of August 2014. Note that we remove meta tags and unnecessary contents, such as lists, titles, template notations, redirect destinations, symbols, and article category information from the Wikipedia dumps. We then convert the resulting data to plain text, which is then tokenized. Stemming is then applied to the data to obtain the words that will be used to train the model. We use a skip-gram model with the following parameters: window size, 5; frequency at which words are cut off, 5; random downsampling threshold, $1e-5$; vector dimensionality, 100.

Note that the proposed method does not require a training dataset for text segmentation. In addition, this method does not rely on randomness. Therefore, results are only calculated according to the average document result once

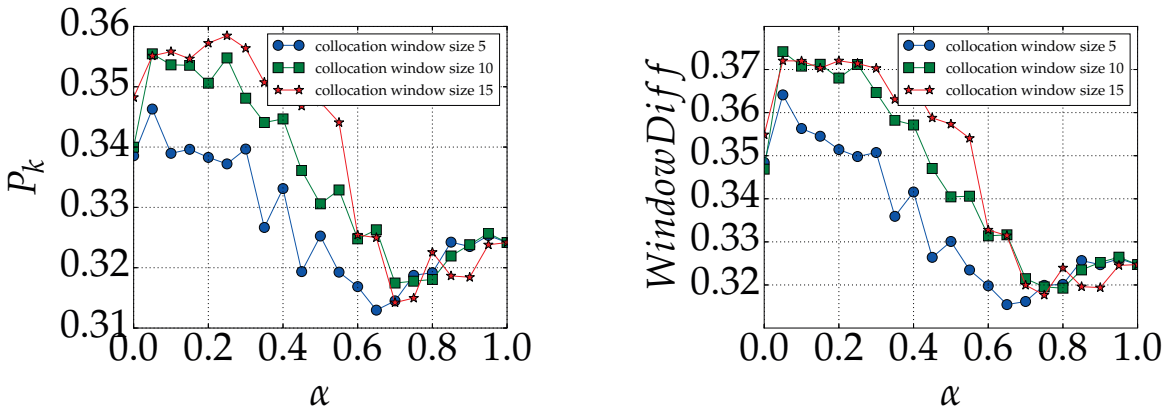

Fig. 5. Relationship between mixture ratio α and collocation window size (clinical dataset with P_k and $WindowDiff$, lower better).

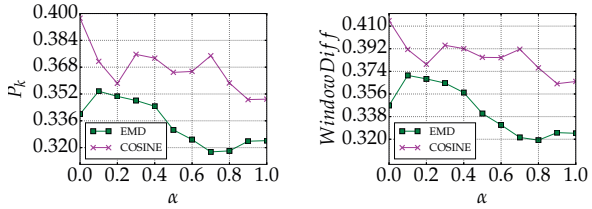

Fig. 6. Comparison of two distance measures with P_k and $WindowDiff$ (collocation window size of 10)

D. Clinical dataset

1) Relationship between Mixture Ratio α and Collocation Window Size: We shift the mixture ratio parameter α and three collocation windows with varying sizes. The results are shown in Figs. 5. When $\alpha = 0.0$, we consider only collocation similarity. For $\alpha = 1.0$, we consider only semantic similarity. Note that P_k and $WindowDiff$ demonstrate better performance at $\alpha = 0.65$ and a collocation window size of 5. Note that these behaviors are similar regardless of collocation window size. The form of the obtained graph resembles a valley, which indicates that either collocation similarity or semantic similarity is insufficient, and the combined similarity is more suitable to our purpose.

2) EMD vs Cosine Distance: To demonstrate the effectiveness of considering the correlation between topical clusters based on EMD, we compared with the cosine distance, which is frequently used to measure the distance between two vectors. Here, the collocation window size was set to 10. As is shown in Fig. 6, EMD outperforms cosine distance for both of P_k and $WindowDiff$. In text segmentation, it is essential to capture correlation of topics and from this perspective, EMD is superior to cosine distance.

3) Comparison with other approaches: To demonstrate the effectiveness of the proposed method, we compared it to several previous approaches; i.e., C99[18], U00[11], LCseg[19], BAYESSEG[2], TopicTiling[4], TSM[5] and NTSeg[6]. The parameter settings and performance of TSM and C99 have been reported by [5]. Those of BAYESSEG, U00, and LCseg have been reported by [2]. Those of NTSeg and TopicTiling have been reported by [6]. TopicTiling, TSM and NTseg require a trained topic model for a given domain; thus, they are domain-dependent. BAYESSEG does not directly require topic model but does use all documents in a dataset to build its segmentation model. C99, U00, and LCseg only use lexical cohesion; thus these approaches are domain-independent. The proposed method requires a word2vec model; however, it can be separated from the target domain and the segmentation task. Therefore, the proposed method is domain-independent. For comparison, the number of boundaries of each document is provided for all approaches, it is gold standard for the text segmentation task.

As is shown in Table I, The proposed method clearly outperforms other domain-independent approaches for both P_k and $WindowDiff$. Therefore, we consider the proposed method a state-of-the-art domain-independent approach. On the other hand, compared to domain-dependent approaches, the proposed method is slightly inferior to TSM and NTSeg with P_k, however, it demonstrates good performance with $WindowDiff$. Therefore, the proposed method is on an even level with domain-dependent approaches. In addition, in the range of $alpha$ between 0.7 and 1.0, proposed method demonstrates good performance with any collocation window size.

TABLE I. COMPARISON OF TEXT SEGMENTATION METHODS FOR CLINICAL DATASET WITH P_k AND $WindowDiff$ (WD) (%). BOLD VALUES DENOTE BEST PERFORMANCE (RANGE, 1%).

Method	P_k	WD	domain-independent
C99 [18]	38.7	39.7	Yes
U00 [11]	37.0	37.6	Yes
LCseg [19]	37.0	38.5	Yes
BAYESSEG [2]	33.9	35.3	No
TopicTiling [4]	31.9	34.7	No
TSM [5]	**30.6**	34.5	No
NTSeg [6]	**30.9**	32.7	No
proposed ($\alpha = 0.65$, collocation window size 5)	**31.3**	**31.5**	Yes

E. News program transcripts

We also applied the proposed method to a real dataset. Note that the number of boundaries of each document is

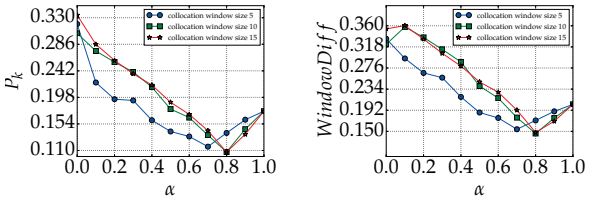

Fig. 7. Relationship between mixture ratio α and collocation window size (news program dataset with P_k and $WindowDiff$). The number of boundaries of each document is not provided.

TABLE II. PERFORMANCE OF PROPOSED METHOD FOR JAPANESE NEWS PROGRAM WITH P_k, $WindowDiff$ (WD), PRECISION (P) AND RECALL (R) (%). COLLOCATION WINDOW SIZE OF 15.

mixture ratio α	P_k	WD	P	R
0.0	33.4	35.4	50.0	31.7
0.8	**10.8**	**14.6**	83.6	**65.2**
1.0	17.5	20.3	**86.9**	48.7

not provided in this examination. The results are shown in the Fig. 7. Both P_k and $WindowDiff$ demonstrated better performance at $\alpha = 0.8$ and collocation window size of 10, 15. As is shown in Table II, combined similarity works in this dataset as well. Fig. 8 shows an example of segmentation, and here, each segment is described briefly with sentence id.

0-26 Losing of Malaysia Airlines Flight 370.
27-40 North Korean abductions of Japanese citizens.
41-57 A case of portable gas stove explosion.
58-70 Weather report.
71-82 The report of a Japanese singer's death.
83-95 A theft by Vietnam Airlines staff.
96-98 Situation of Ukraine.
99-101 Opening of new road for Tokyo Olympic.
102-104 The damage of Tsunami.
105-107 An examination result announcement for a theater.
108-133 High school baseball.
134-135 Japan professional football league.
136-139 Weather report.

Most boundaries were predicted correctly; however, the boundaries between 101 and 102, 133 and 134 were missed because those segments contain too many short sentences and few words. Fig. 9 shows the word stream around the boundary for the topical cluster id shown in Fig. 8. Here, a reference boundary exists between sentences 26 and 27, and the topic shifts to "North Korean abductions of Japanese citizens" from "Losing of Malaysia Airlines Flight 370". From a topical cluster id perspective, these sentences appear to be unrelated; however, considering the distance between clusters, sentences 25 and 26 are similar, and are the same as sentences 27 and 28. Therefore, while calculating the distance between sentences with cosine distance decreases performance, EMD can calculate the appropriate distance. Generally, rough clusters generate clear topics, but reduce the boundary detection rate. Thus, there is a trade-off between number of clusters and text segmentation performance. However, the proposed method is compatible with both easily understanding topics and boundary detection rate.

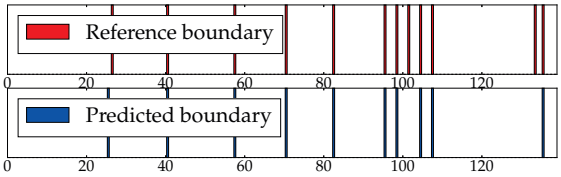

Fig. 8. Example of segmentation with sentence id (Japanese news program).

Sentence 25. single(56), flight_recorder(61), location(64), notify(60) , sound_wave(67), make(23), send(25), power_source(74)

Sentence 26. power_source(74), run_out(23), search(63), Malaysia(4), airplane(61), trace(63), find(66), search(63), fight(39)

Sentence 27. Japan(43), North_Korea(8), government(49), consultation(42), Peking(3), resume(22), Japan(43), abduction(15), investigation(65), abduction(15), solve(49), get_better(49), concrete(56), approach(42), request(49), plan(42)

Sentence 28. consultation(42), resume(22), abduction(15), damage(50), family(32), on_the_street(25), blood_relation(32), early_stage(42), go_home(21), appeal(49)

Fig. 9. Example of word stream with topical cluster id around boundary between 26 and 27. Words of the similar color indicates that the distance between topical clusters is less than 0.3 (max 1.0) except black. These words are translations of Japanese words; some words consisting of more than two words are a single word in Japanese.

IV. CONCLUSION

To extract detail meta information of data, we have proposed a domain-independent unsupervised text segmentation method, which is applicable to even if unseen single document. The proposed method applies combined similarity between words consisting of semantic similarity obtained from word2vec and collocation similarity obtained from within the document to clustering. The proposed method then, calculates the distance between sentences based on EMD between the frequencies of the obtained topical clusters. This proposed method clearly outperforms state-of-the-art domain-independent approaches and obtains equal performance with state-of-the-art domain-dependent approaches, such as those that use topic modeling. The proposed method is applicable to segment a document and a moving image data with transcripts such as a movie and a TV program. After segmentation of these data, for each segment, by measuring term frequency, various meta information such as flow of topics, atmosphere of each scene and so on is extracted.

ACKNOWLEDGEMENT

We thank Professor Yukio Ohsawa for insightful suggestions. Our method was inspired by his research of the KeyGraph and the data crystallization. And we thank to AOARD program for financial support.

References

[1] M. A. Hearst, "Texttiling: Segmenting text into multi-paragraph subtopic passages," *Comput. Linguist.*, vol. 23, no. 1, pp. 33–64, Mar. 1997. [Online]. Available: http://dl.acm.org/citation.cfm?id=972684.972687

[2] J. Eisenstein and R. Barzilay, "Bayesian unsupervised topic segmentation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '08. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 334–343. [Online]. Available: http://dl.acm.org/citation.cfm?id=1613715.1613760

[3] H. Misra, F. Yvon, J. M. Jose, and O. Cappe, "Text segmentation via topic modeling: An analytical study," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ser. CIKM '09. New York, NY, USA: ACM, 2009, pp. 1553–1556. [Online]. Available: http://doi.acm.org/10.1145/1645953.1646170

[4] M. Riedl and C. Biemann, "Topictiling: A text segmentation algorithm based on lda," in *Proceedings of ACL 2012 Student Research Workshop*, ser. ACL '12. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 37–42. [Online]. Available: http://dl.acm.org/citation.cfm?id=2390331.2390338

[5] L. Du, W. Buntine, and M. Johnson, "Topic segmentation with a structured topic model," in *2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Atlanta, USA, June 2013, p. 11.

[6] S. Jameel and W. Lam, "An unsupervised topic segmentation model incorporating word order," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '13. New York, NY, USA: ACM, 2013, pp. 203–212. [Online]. Available: http://doi.acm.org/10.1145/2484028.2484062

[7] D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, p. 2003, 2003.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[9] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *CoRR*, vol. abs/1310.4546, 2013. [Online]. Available: http://arxiv.org/abs/1310.4546

[10] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, 2007. [Online]. Available: www.psi.toronto.edu/affinitypropagation

[11] M. Utiyama and H. Isahara, "A statistical model for domain-independent text segmentation," in *In Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, 2001, pp. 491–498.

[12] P. Fragkou, V. Petridis, and A. Kehagias, "A dynamic programming algorithm for linear text segmentation," *J. Intell. Inf. Syst.*, vol. 23, no. 2, pp. 179–197, Sep. 2004. [Online]. Available: http://dx.doi.org/10.1023/B:JIIS.0000039534.65423.00

[13] M. F. Porter, "Readings in information retrieval," K. Sparck Jones and P. Willett, Eds. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997, ch. An Algorithm for Suffix Stripping, pp. 313–316. [Online]. Available: http://dl.acm.org/citation.cfm?id=275537.275705

[14] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases," in *Proceedings of the Sixth International Conference on Computer Vision*, ser. ICCV '98. Washington, DC, USA: IEEE Computer Society, 1998, pp. 59–. [Online]. Available: http://dl.acm.org/citation.cfm?id=938978.939133

[15] D. Beeferman, A. Berger, and J. Lafferty, "Statistical models for text segmentation," *Mach. Learn.*, vol. 34, no. 1-3, pp. 177–210, Feb. 1999. [Online]. Available: http://dx.doi.org/10.1023/A:1007506220214

[16] L. Pevzner and M. A. Hearst, "A critique and improvement of an evaluation metric for text segmentation," *Comput. Linguist.*, vol. 28, no. 1, pp. 19–36, Mar. 2002. [Online]. Available: http://dx.doi.org/10.1162/089120102317341756

[17] M. Scaiano and D. Inkpen, "Getting more from segmentation evaluation," in *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, ser. NAACL HLT '12. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 362–366. [Online]. Available: http://dl.acm.org/citation.cfm?id=2382029.2382078

[18] F. Y. Y. Choi, "Advances in domain independent linear text segmentation," in *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, ser. NAACL 2000. Stroudsburg, PA, USA: Association for Computational Linguistics, 2000, pp. 26–33. [Online]. Available: http://dl.acm.org/citation.cfm?id=974305.974309

[19] M. Galley, K. McKeown, E. Fosler-Lussier, and H. Jing, "Discourse segmentation of multi-party conversation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ser. ACL '03. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003, pp. 562–569. [Online]. Available: http://dx.doi.org/10.3115/1075096.1075167