NTCIR-15 QA Lab-PoliInfo2 Dialog Topic Detection based on Discussion Structure Graph

Yuya HIRAI
Faculty of Information and Communication Engineering
gp17a108@oecu.jp

Yo AMANO
Faculty of Information and Communication Engineering
gp17a003@oecu.jp

Kazuhiro TAKEUCHI
Faculty of Information and Communication Engineering
takeuchi@osakac.ac.jp

ABSTRACT

This paper proposes a dialog summarization method using graph representation that shows the target discussion structure. Firstly our proposed method extracts words related to the opinion and position of each participant based on co-occurrence. Then, LDA (Latent Dirichlet Allocation) is applied to classify position and opinion words, while we determine the number of topics needed for LDA as its parameter by hierarchical clustering using co-occurrence frequency. Finally, topic phrases as output are generated by using the dependency structure analysis.

TEAM NAME

TKLB

SUBTASKS

Dialog Topic Detection, LDA (Latent Dirichlet Analysis), Discussion structure Graph

1 INTRODUCTION

Facing the problem caused by the new coronavirus infection (COVID19), it is more important for the national and local governments to communicate with the residents quickly and appropriately. QALab-PoliInfo-2[2] performs the tasks (stance classification, dialog summarization, entity linking, and topic detection) to explore the information technology in these situations. There are already two existing media of communication between them: one version issued by local governments conveys the contents discussed in the parliament in an easy-to-understand manner, but it takes time to publish because it is created manually. The other version quickly conveys the contents of representative questions and general questions from members who attended the discussion, but there is room for improvement in terms of the ease of understanding the agenda and issues.

In this paper, we propose a way to show an equivalent representation of the former version by an automated processing of the latter based on a graph where nodes correspond to the word uttered by the discussion participants. We assume the proposed graph structure that shows conflicting opinions and positions in the discussion. In other words, our purpose in this paper is to show the conflict points in a discussion. For making the structure, we employ LDA to identify the words to attention.

2 DISCUSSION STRUCTURE IN DISCUSSION

Our primary strategy for summarizing is not based on text coherence of discussion. Our proposed method focuses on expressing the differences of word use between its participants. Figure 1 shows our graph representation of discussion, which we refer to as discussion structure graph in this paper. In this figure, the round nodes show the utterances expressed in natural language, the square nodes show the participants, and each link shows the relationship between which participant utters a particular word. As shown in this figure, this discussion has a common topic or subject, but at the same time has different positions among the participants. Specifically, we can distinguish the nodes into two classes with the adjacency relations among them in this graph representation: one is the subject or topic of the discussion (gray nodes), and the other is the participant's position or opinion (yellow and green nodes).

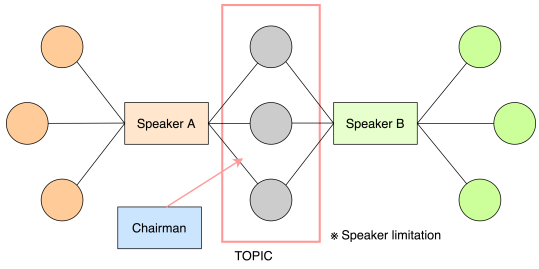

Figure 1: Argument structure graph

3 DATA

As the preprocessing for summarizing the transcripted data from the QA Lab-PoliInfo-2 Committee, we divide the data into hierarchical units that we assume to correspond to written language units. The largest unit corresponds to the segment that each speaker argues in the discussion. The next largest unit is regarded as a paragraph, and the smallest unit corresponds to a sentence in written language.

In the summarization process, we use two packages of free software. One is KHcoder[1], a tool that efficiently performs sophisticated analyses (such as correspondence analysis, cluster analysis, multidimensional scaling, self-organizing map, co-occurrence network, machine learning) by GUI operation. The other is Mallet[4], which we use for performing LDA.

As processing by KHcoder, it conducts morphological analysis to the input data using Chasen[3]. In this process, we exclude the words shown in Table 1 as stop words.

Table 1: List of stopword

KHcoder	品詞名	POS
名詞 B	普通名詞	がん, みずから, きっかけ
動詞 B	動詞	よい, ふさわしい
形容詞 B	形容詞	する, ある, ない, いたす
副詞 B	副詞	かつて, とりわけ, これから
否定助動詞	助動詞	ない, ん, ぬ
形容詞 (非自立)	形容詞	やすい, ほしい, ない, いたす

4 PROPOSED METHOD

4.1 Discussion structure Graph

Figure 2 shows an example of the exacted discussion structure graphs from the transcription data. For any time interval of a given discussion, one can make this kind of graph, where it shows the words and speakers as nodes and who said the words as links. The given duration to Figure 2 is a specific day. Each ellipse in Figure 2 shows three structures similar to that shown in Figure 1 that we assume to represent conflicts among participants in the previous section.

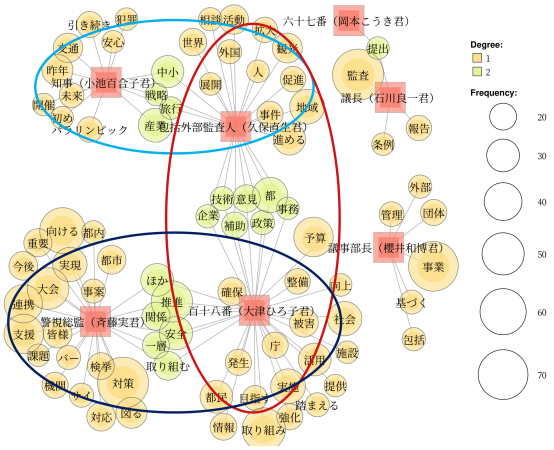

Figure 2: An example of exact argument structure graph from the discussion

4.2 Clustering the words

As explained so far, our proposed method detects topics based on a graph representation with word nodes. To find topics expressed in different words in all given data, we conduct word clustering that might be known as the most fundamental method.

Figure 3 shows a part of the results of a tree of clustering using the Ward method. In most hierarchical clustering methods, including the clustering shown here, the criteria for whether a given pair is similar or not is estimated using metrics based on the co-occurrence frequencies in an individual unit of text or discourse.

Word co-occurrence can be a fundamental metric, but we regard the word similarity based on the raw co-occurrence frequencies as a rough guideline. In concrete, we use this simple clustering to estimate the number of topics in the argument. In Figure 4, the horizontal axis shows the process of integrating the clusters from leaves to the root in the hierarchical tree, and the number of clusters is plotted for each integration until the number of clusters becomes one. Since the most massive change in the slope of plots occurred when the number of clusters was six in Figure 4, we regard that all of the given discussions can consist of six topics. We set the number of topics specified for LDA to six. (In other words, from this examination, we assume that six topics in the LDA model generate the given discussions through 8 days.)

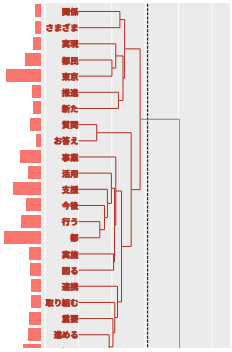

Figure 3: A part of hierarchical clustering tree

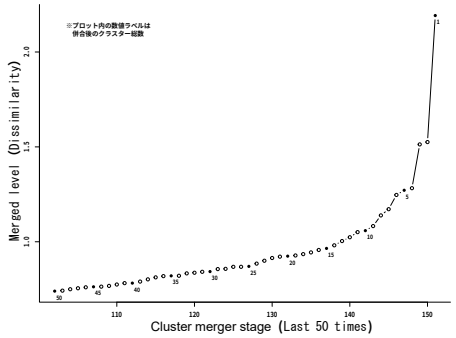

Figure 4: Hierarchical process of integrating clusters

4.3 Words for each topic

We employ a generative model that assumes a probabilistic mechanism of how participants generate their utterances, assuming the topics to discuss. For the purpose, we use Mallet: a Java-based package for machine learning applications that can perform statistical natural language processing, document classification, clustering, topic modeling, etc. Specifically, we estimated that each word contributes to constructing a certain topic using topic modeling analysis in Mallet[4], which employs LDA (Latent Dirichlet Allocation).

Table 2 shows the results of LDA. For the number of topics, we used six, which results from the discussion in the previous section. Using the word list in Table 2, we extract each participant's position, and opinion nodes strongly related to constructing a specific topic from the discussion structure graph explained in Section 4.1.

Table 2: Word-topic list acquired with LDA

Topic Number	topic words
1	東京 大会 取り組み 実現 都民 企業 重要 活用 都市 社会 推進 支援 事業 今後 新た さまざま 戦略 スポーツ 連携 向上
2	感染 コロナ 新型 対策 支援 ウイルス 事業 拡大 対応 体制 検査 必要 防止 医療 対し 都民 状況 今後 確保 影響
3	支援 学校 医療 教育 子供 相談 区市 児童 事業 対応 実施 今後 取り組み 必要 保護 施設 おけ 障害 活用 地域
4	知事 伺い 考え 見解 求め 必要 東京 都民 党 議会 日本 予算 小池 病院 要望 学校 でき 負担 声 つ
5	整備 対策 地域 公園 事業 計画 推進 災害 防災 今後 都市 取り組み 道路 安全 交通 進め 重要 連携 質問 施設
6	委員 報告 議案 東京 令和 意見 審査 結果 条例 請願 議会 関する 決定 陳情 予算 議長 付託 良一 石川 財政

4.4 Topic phrase generation

Our output is based on the differences discovery of differences in each participant's positions and opinions through the graph structure analysis and the weighting of words with LDA. Specifically, we extract the most weighted position or opinion node of each participant in a discussion structure graph.

For example, consider Figure 5 that is an enlarged graph showing the words used by one of the participants shown in Figure 2. In the figure, yellow nodes indicate the words used frequently by this participant, and there are many words (The more frequently the words indicated by a node, the larger the node size). Looking up with Table 2, which is the result of LDA; among these many words, one can find the word ' 取り組み' as the highest weight.

In this way, we focus on the word ' 取り組み' of the participant and extract the participant's sentence that firstly appears with the word in the discussion. We generate topic phrases for the target participant based on the dependency structure of the sentence analyzed with Cabocha[5]. Specifically, we remove the punctuation and the trailing morphemes in segments of dependency structure to generate its proper topic phrase. The segment connected by an arrow has a dependency from the segment to the segment. In the figure, the segment ' 制度周知の' depends on the segment ' 取り組みは' that includes the focused word. With our manually designed scripts that reforms the last segments (removing the postposition in this case), the topic phrase ' 制度周知の取り組み' is generated as an output.

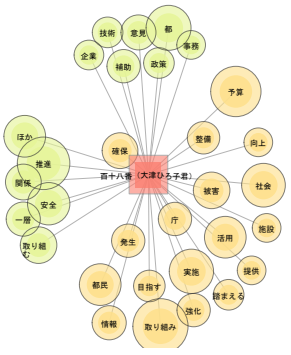

Figure 5: Enlarged graph of a participant

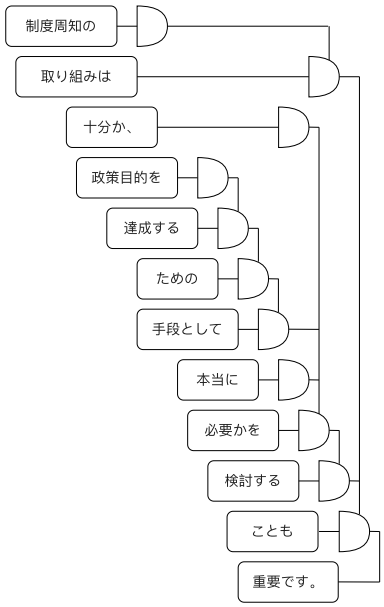

Figure 6: An example of dependency structure of a sentence with ' 取り組み'

5 CONCLUSION

This paper proposed extracting topic phrases based on the co-occurrence graph and finding conflicts of opinions and positions among participants. In the proposed method, the graph (we refer to it as discussion structure graph) generated with the information who spoke each word, defines the words that characterize each participant's opinion and position based on their adjacent relations. We also conducted a topic analysis with LDA to grasp the broader topics in all of the given discussion. The essential sentence of each participant was detected using the results of LDA. In the output process, the dependency structure was used for the detected essential sentences, and the opinions and positions of each participant were generated as topic phrases by the transformation rules that we created manually.

REFERENCES

[1] Koichi Higuchi. 2016. A two-step approach to quantitative content analysis: KH Coder tutorial using Anne of Green Gables (Part I). *Ritsumeikan Social Science Review* 52, 3 (2016), 77–91.

[2] Yasutomo Kimura, Hideyuki Shibuki, Hokuto Ototake, Yuzu Uchida, Keiichi Takamaru, Madoka Ishioroshi, Teruko Mitamura, Masaharu Yoshioka, Tomoyoshi Akiba, Yasuhiro Ogawa, Minoru Sasaki, Kenichi Yokote, Tatsunori Mori, Kenji Araki, Satoshi Sekine, and Noriko Kando. 2020. Overview of the NTCIR-15 QA Lab-PoliInfo-2 Task. *Proceedings of The 15th NTCIR Conference* (12 2020).

[3] Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano, Hiroshi Matsuda, Kazuma Takaoka, and Masayuki Asahara. 1999. Japanese morphological analysis system ChaSen version 2.0 manual. *NAIST Techinical Report* (1999).

[4] Andrew Kachites McCallum. 2002. MALLET: A Machine Learning for Language Toolkit. (2002). http://mallet.cs.umass.edu.

[5] Yuji Matsumoto Taku Kudo. 2002. Japanese Dependency Analysis using Cascaded Chunking. (2002), 63–69.