Detecting Evolution of Keyphrases from Wikipedia Edit History

Qianhui WEI † Zihang CHEN ‡ and Mizuho IWAIHARA ‡

† Graduate School of Information, Production and Systems, Waseda University Kitakyushu, Japan

E-mail: † wqh@toki.waseda.jp, ‡ chenzihang @waseda.jp, ‡ iwaihara@waseda.jp

Abstract Wikipedia stores edit history which records revisions of each article. From the edit history, we can extract a great amount of information on how real world events happened and evolved. A significant event is often given a name by the public or mass media, which is also reflected onto Wikipedia articles. Keyphrases of a sequence of events may be changed or new keyphrases are spawn as the events evolve, which we regard as a topic transition. TextRank is an effective algorithm to extract keyphrases from word co-occurrence graphs, but it does not consider temporal trends in text stream. In our previous work, we proposed TextRank_nfidf which considers temporal changes of word co-occurrences. In this paper, we discuss detecting evolving keyphrases along history. We apply our temporal keyphrase extraction to several Wikipedia categories and articles, to track attention of edits transferring as events develop. We examine long-term edit trends of relevant phrases to discover transfer of editing attentions. Each word has its own bursting periods, which is captured as a high-score node in our graph. Bursting periods of words may overlap, and if one word is later succeeded by another word, such a change can be regarded as a topic transition. We discuss methods for detecting topic transitions from temporal changes of keyphrases.

Keyword edit history, keywords extraction, TextRank, keyphrases graph

1. Introduction

Wikipedia is now one of the most prominent encyclopedia on the Internet. All the revisions of each article and its related information are stored in Wikipedia edit history. Revisions are often triggered by real world events. Editors notice about events from news articles, social media, or some other sources, and select new facts to be added into Wikipedia articles. So compared with news articles, Wikipedia edit history shows how an event happens and evolves in a more summarized and organized manner.

A number of work have been focuses on detecting and analyzing bursts of text stream [2][3]. In our previous work [2], we proposed TextRank_nfidf to extract keyphrases that can represent the topics of document streams in a given collection of articles. Our algorithm shows superior performance over TextRank. We contrasted quality of extracted phrases with Google Trends, and the result shows that the keyphrases we extracted are well representing real world events in burst periods.

In this paper, we apply TextRank_nfidf to several Wikipedia categories to track attention of edits transferring as events develop. We examine long-term edit trends of relevant phrases to discover transfer of editing attentions. Each word has its own bursting periods, and the bursting word is captured as a high-score node in our graph. Bursting periods of words may overlap, and if one word is later succeeded by another word, such a change can be regarded as a topic transition. The keyphrases node graphs represent continuously changed topic of selected articles or categories. If we slightly adjust the articles in the article set, the main topic reflected in the node graphs will also change. It is crucial for finding articles that are related with a given topic in order to dig more related keyphrases about it. We discuss the necessary to design an algorithm for selecting articles for a given topic. Section 2 covers related work. Section 3 is an introduction of TextRank_nfidf proposed in [2]. Section 4 is the framework and requirements for detecting keyphrase transitions. Section 5 is experiments and discussions. Section 6 is a conclusion.

2. RELATED WORK

Liu, Ruoran, et al. [4] discussed utilizing temporal information, topic information to mine evolution phases of hot events. They categorize the phases of occurrences into development, climax, decline, and ending during the lifespan of hot events.

Inspired by PageRank [8], Mihalcea and Tarau [6] proposed TextRank, which extracts significant keywords by co-occurrence relationship between words. TextRank constructs an edge weighted graph and gives ranking scores based on the PageRank algorithm. Bellaachia and Al-Dhelaan [1] added node weights to TextRank, which improves the precision in extracting keywords in document sets. But these keyword extraction methods do not consider temporal information of edits on the documents. In our work, we reconsider node weights and

assign smoothed edit activity levels as node weights of phrases.

In 2008, J.P. Herrera et al. tackled the problem of finding and ranking the relevant words of a document by using statistical information referring to the spatial use of the words [5]. Shannon's entropy of information was used for automatic keyword extraction. The randomly shuffled text was used as a standard and the various measures used in the original document text were normalized by corresponding measures of random text.

P. Carpena et al. proposed to automatically extract keywords from literary texts through a generalization of the level statistics analysis of quantum disordered systems [7]. They consider frequencies of the words along with their spatial distribution along the text, and is based on the observation that important words are significantly clustered whereas irrelevant words are distributed randomly in the text. No reference corpus is needed in this approach and it is especially suitable for single documents for which no priori information is available.

3. TextRank_nfidf

In this section, we describe TextRank_nfidf [2] which is used for detecting bursty keyphrases from revision sequences.

3.1 Keyphrase Extraction based on TextRank_nfidf

TextRank is a well-known method to extract keyphrases. It is inspired from PageRank for web page scoring. TextRank utilizes co-occurrence relationship between words to construct an edge weighted graph. We assume that a set S of articles, called the *target article set,* or simply *target set*, is given. Each article in S is a sequence of revisions. We call the text difference of two consecutive revisions of one article a *revision delta*.

We apply part-of-speech (POS) Tagger of Apache Open NLP to divide each sentence of revision deltas into chunks, where chunks containing noun POS tags are regarded as phrases. We construct a graph such that each node is labeled with an extracted phrase, and there is an edge (V_i, V_j) with weight w_{ji}, from node V_i to node V_j, if the phrases of V_i and V_j co-occur in a window of maximum M words in the text. The weight w_{ji} is equal the co-occurrence count of V_j and V_i in this window.

The definition of the score function of TextRank is shown in (1).

$$WS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \quad (1)$$

Here, $WS(V_i)$ is the rank score of node V_i, d is the damping factor usually set to 0.85, and w_{ji} is the edge weight. $In(V_i)$ is the set of nodes that point to V_i, $Out(V_j)$ is the set of nodes that node V_i point to. TextRank only takes edge weights into account. This leads to a situation such that common words which appear many times in the associated text have a high rank score.

Chen Zihang, et al. [2] proposes TextRank_nfidf which reduces the rank of common words by adding a node weight $W(V_i)$ to original TextRank. The score function is defined as:

$$WS(V_i) = (1-d) * W(V_i) + d * W(V_i) * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \quad (2)$$

Here, the node weight is defined as $W(V_i)_{nfidf}$:

$$W(V_i)_{nfidf} = NF(V_i) * \log_2 \frac{L}{DF(V_i)} \quad (3)$$

Here, L is the number of articles appeared in the history of the given target article set S, and $DF(V_i)$ is the number of articles containing phrase V_i. The number of articles L is changing over time. But for simplicity, L is set to be the number of all the articles that ever appeared.

The net frequency (NF) is to capture net edit activities of each phrase, defined below:

$$NF_t(V_i) = max\left\{0, \sum_{k=1}^{L} \sum_{j \in rev(k,t)} \frac{RTFD_{j,k}(V_i)}{\sum_{k=1}^{L} |rev(k,t)|}\right\} \quad (5)$$

Here, $rev(k,t)$ is the set of revision IDs of the k-th article, created in the t-th week. $NF_t(V_i)$ represents the net frequency of term V_i in the t-th week before the current time. $RTFD_{j,k}(V_i)$ measures the difference of frequency of phrase Vi between j-th revision and $(j-1)$-th revision in the k-th article. Then the exponential moving average is applied to smooth the net frequency, as below:

$$NF(V_i) = \frac{NF_1(V_i) + \eta * NF_2(V_i) + \eta^2 * NF_3(V_i) + \eta^3 * NF_4(V_i) \ldots}{1 + \eta + \eta^2 + \eta^3 + \ldots}$$

Here, $NF_1(V_i)$ is the net frequency of node V_i at the current week, $NF_2(V_i)$ is the net frequency of node V_i at the last week and so on. The coefficient $0 \leq \eta < 1$ gives the damping factor.

With the score function of (2), the TextRank algorithm gives scores to the nodes of the graph.

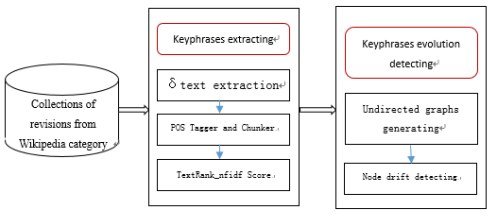

Fig. 1. Framework of our system

4. Detecting topic trajectory

Our objective is to track attention of edits transferring as events develop. First we extract phrases which can represent bursts in revision streams using TextRank_nfidf. We regard a phrase scored high by TextRank_nfidf represents a topic, and call it a *keyphrase*. Bursting periods of keyphrases may overlap, and if one keyphrase is later succeeded by another keyphrase, such a change can be regarded as a topic transition. We discuss algorithms for detecting topic transitions from temporal changes of keyphrases.

Each phrase has its own bursting periods, which is captured as a high-score node in the graph. New keyphrases may spawn from existing keyphrases, as time lapses. We can detect such transition of keyphrases by comparing temporal changes of the phrase graphs. We can formalize criteria of topic transition by the following conditions:

a) Two phrases p_1 and p_2 have overlapping burst periods,
b) p_1 and p_2 are connected by an edge in a phrase graph, and
c) the first burst of p_2 is after the first burst of p_1.

Condition (b) means that p_1 and p_2 co-occur in a window of maximum M words in one article. Since the magnitude of each burst is relative to the scale of articles, we have to carefully determine criteria for topic transition. The following situations have to be taken into account.

1) The target article set S defines the context of a burst. If S is chosen as a Wikipedia category, we will observe bursts regarding the topic of the category.
2) A keyphrase p may not be detected if S contains too many articles and a burst of S can be hidden among other popular phrases. A compact set S in which p shows a clear burst can be regarded that the articles in S are influenced by the burst of p.
3) If the target article set S becomes larger, computation cost for TextRank score grows nonlinearly.
4) A keyphrase p may incur frequent edits but later a different name can be used, which should be detected as topic transition. However, such transition can be outside of the target set S. Thus, to detect transition we have to update the target set.
5) Keyphrases of a target set S are representing topics of S. Also, from a bursting keyphrase p. topic transition to one of these keyphrases can happen.
6) A quite significant keyphrase may emerge and widely occur in articles of various categories, which may not accompany with appropriate links to the origin of the keyphrase. In such a case, the target set S has to be reduced to more specific categories to reflect a more focused topic rather than a large-scale boom.

The above discussion indicates that the selection and update of the target article set S is crucial for finding quality keyphrases that can capture topic transition and evolution. In future work, according to the above requirements we plan to design an algorithm for updating the target set S along the timeline.

5. Experiments

1) Article set

We collected all the revisions from Wikipedia category “Russian interference in the 2016 United States elections.” The collected revisions were created between 2015/10/01 and 2018/10/01 (totally 157 weeks). There are 28 articles and 17845 revisions.

2) Keyphrase extraction and evolution

Fig.1 shows the framework of our system. We apply TextRank_nfidf to our dataset. Fig.2 shows the weekly TextRank_nfidf score of six significant phrases “Clinton”, “Trump”, “Russia”, “Dossier”, “Comey”, and “Mueller” in our dataset. It shows how rank score of a word changes as events evolve.

In early March 2015, Hillary Clinton was revealed that she used private email server for official communications. This is the beginning of a chain of Russian interference exposure. From week 1 to around week 60 the articles are edited regarding Clinton’s email controversy. Starting from week 70, Russian Government was reported to direct email hacking to interrupt U.S. election. TextRank scores of keyphrases “Trump” and “Russia” rise rapidly. Soon after that, new article “Trump-Russia dossier” was created and the word “Dossier” first appears in the dataset. At week 82, Trump dismissed FBI Director James Comey. Mueller was appointed as special counsel overseeing investigation into Russian interference. TextRank score of word “Comey” reaches its highest at around week 90 and word “Mueller” at around 106. Bursting periods of “Comey” and “Mueller” overlap, and “Comey” is later succeeded by “Mueller”. It shows that the public’s attentions are soon transferred from Comey’s dismissal to Mueller’s investigation.

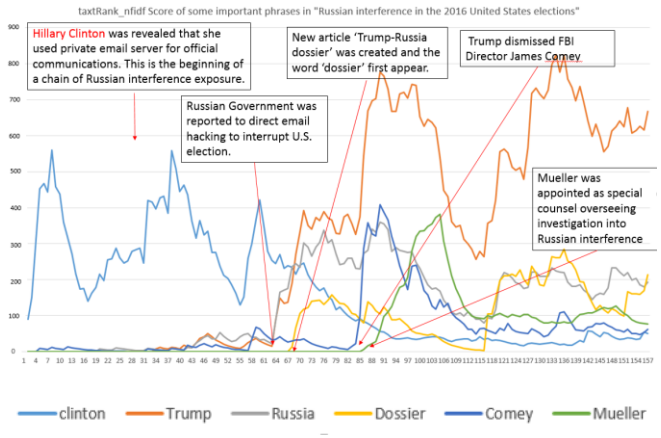

Fig.2. Weekly TextRank_nfidf score of some important words

We visualize keyphrases using undirected graphs where keyphrases are represented as nodes and relationship between keyphrases are represented by edges. The size of nodes represents TextRank_nfidf score. Phrases with high score are represented as large nodes. Fig.3 (a)-(d) show graphs of weeks 60, 70, 90, and 106.

Clinton's email controversy is the beginning of a series of Russian inference scandal exposure. At week 60, people only talks about Clinton's email controversy (Fig.3(a)). At week 70, the connection between Trump and Russia was reported to the public (Fig.3(b)). Soon, Trump attracted the editors' attention and the node "trump" becomes larger than "clinton." New significant nodes like "putin," "Russia," "dossier" appear. At around week 90, Comey, the director of the FBI was dismissed by Trump (Fig.3(c)). In this week, "comey," "dismissal" become hot nodes. Then, Mueller was appointed as special counsel of Russian interference. Node "mueller" becomes larger than "comey" at week 106 due to the transfer of public attention (Fig.3(d)).

The nodes "mueller" and "comey" each has its own bursting periods, which overlap with one another. "comey" is later succeed by "mueller" in a topic transition. The focus of the topic transits from Comey's dismissal to Mueller's nomination. This is a typical example of topic transition.

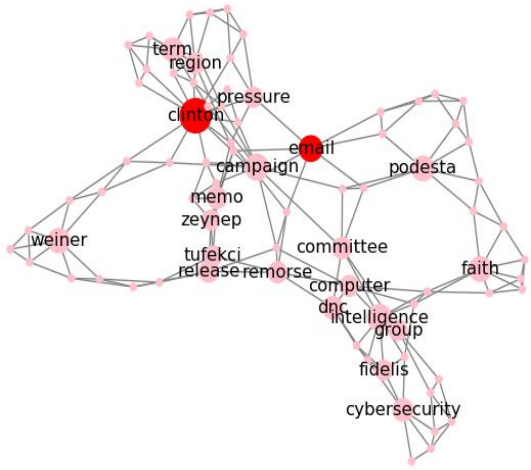

Fig. 3(a) Keyphrase graph of week 60

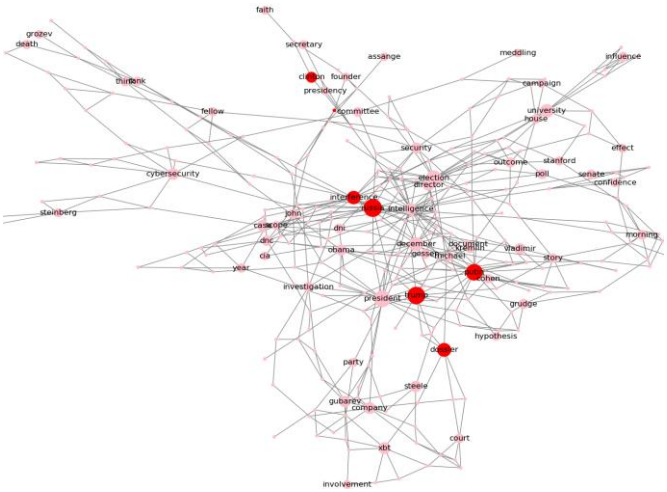

Fig. 3(b) Keyphrase graph of week 70

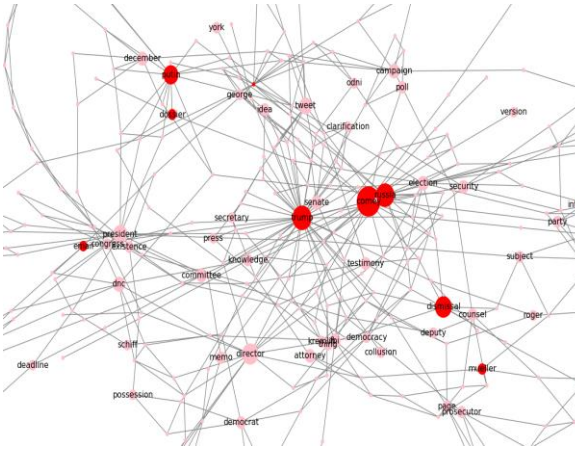

Fig. 3(c) Keyphrase graph of week 90

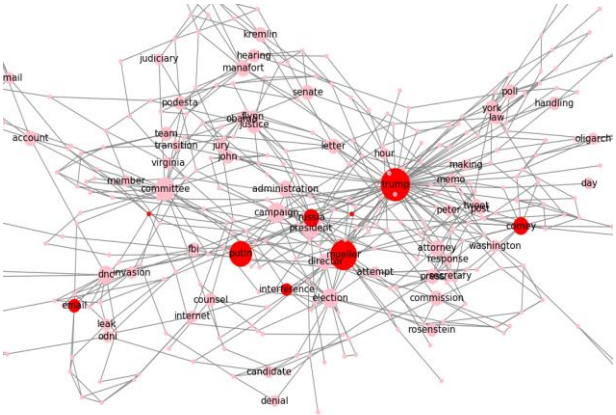

Fig. 3(d) Keyphrase graph of week 106

3) Renew article set

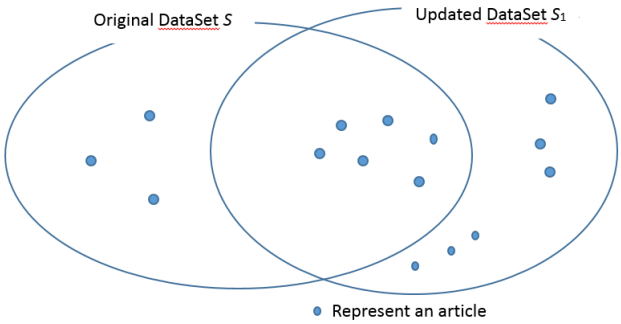

Fig. 4 Updated article set

In order to find more related events and keyphrases about 'comey' at around week 90 (corresponding with the event "dismissal of James Comey"), we renewed our article set by adding more articles related with 'comey' and deleting articles that do not contain word 'comey' (Fig.4). We use TextRank_nfidf to calculate score of keyphrases again in the new article set and get a new node graph which keep 'comey' as the central.

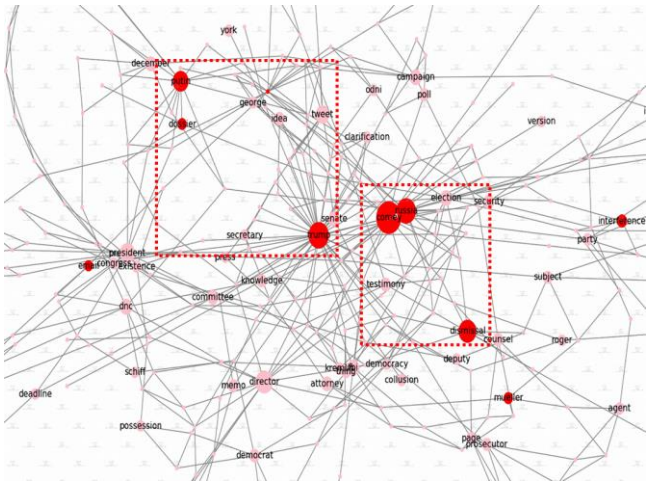

Fig. 5(a) node graph at week 90 of the original article set

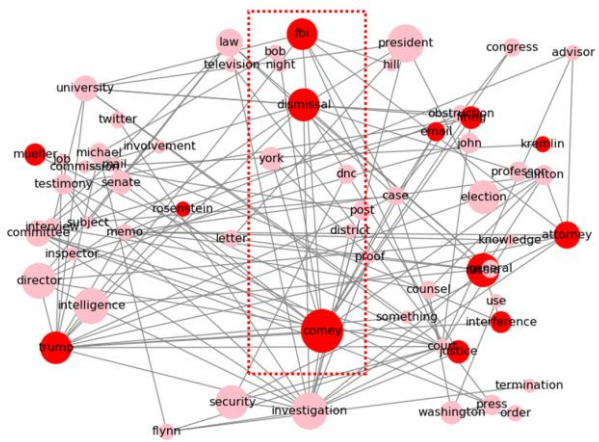

Fig. 5(b) Node graph at week 90 of the original article set

Fig.5(a) is the node graph at week 90 of the original article set. The main event in this graph is Trump-Russia dossier. Comey's dismissal appears in this graph. But many related keyphrases are not included. Fig. 5(b) is the node graph at week 90 of the renewed article set. The event of Comey's dismissal is clearly captured. Many related keyphrases about Comey's dismissal are shown. The neighbor nodes of 'comey' include 'rosenstein,' 'congress', 'russia,' 'dismissal,' 'firing,' 'investigation,' 'interference' and so on. It shows that if we slightly adjust the article set, the topic of the node graph and keyphrases will also change. The original article set is from a large category, where many events are captured in the node graph. If we want to clearly capture the event that we are concerned about, we can add more articles or delete articles regarding the phrase to renew the article set. The updated article set should contain topics more closely related to the selected keyphrase and burst.

6. Conclusion and Future work

In this paper, we utilize TextRank_nfidf to extract keyphrases and discuss detecting evolving keyphrases along history. We show weekly results of ranked keyphrases and explain how the scores are affected as events develop. We visualize keyphrases and their relationship using keyphrase graphs. Node graphs are closely reflecting the topics of given article set. We find that if we slightly adjust the article set, more keyphrases related with given topic can be detected. In the future, we will discuss an algorithm for article set update.

References

[1] Bellaachia A, Al-Dhelaan M, "NE-Rank: A Novel Graph-Based Keyphrase Extraction in Twitter," In Proc. Conf. Web Intelligence & Intelligent Agent Technology, 2012, pp. 372-379.

[2] Chen, Z. and M. Iwaihara, "Detection of Bursty and Significant Keyphrases from Wikipedia Edit History," Proc. IEEE Int. Conf. Big Data and Smart Computing (BigComp2019), Feb. 2019.

[3] Christiansen, L., Schimoler, T., Burke, R., & Mobasher, B, "Modeling topic trends on the social web using temporal signatures," In Proc. 12th Int. Workshop on Web information and data management, Nov 2012, pp. 3-10.

[4] Liu, Ruoran, et al, "Mining phase evolution for hot topics: A case study from multiple social media platforms," IEEE Int. Conf. IEEE, Systems, Man, and Cybernetics (SMC), 2017, pp. 2814-2819.

[5] J.P. Herrera, P.A. Pury, —Statistical keyword detection in literary corpora‖, The European physical journal, 2008

[6] Mihalcea R, Tarau P, “TextRank: Bringing order into text,” In Proc. Conf. Empirical Methods in Natural Language Processing, 2004.

[7] P. Carpena et al., ―Level statistics of words-Finding keywords in literary texts and symbolic sequences‖, Physical Review E, 79, 03512(R), 2009

[8] Page, Lawrence, et al, “The PageRank citation ranking: Bringing order to the web,” Stanford InfoLab, 1999.