Discovering Story Chains: A Framework Based on Zigzagged Search and News Actors

Cagri Toraman
Bilkent Information Retrieval Group, Computer Engineering Department, Bilkent University, Ankara 06800, Turkey. E-mail: ctoraman@cs.bilkent.edu.tr

Fazli Can
Bilkent Information Retrieval Group, Computer Engineering Department, Bilkent University, Ankara 06800, Turkey. E-mail: canf@cs.bilkent.edu.tr

A story chain is a set of related news articles that reveal how different events are connected. This study presents a framework for discovering story chains, given an input document, in a text collection. The framework has 3 complementary parts that i) scan the collection, ii) measure the similarity between chain-member candidates and the chain, and iii) measure similarity among news articles. For scanning, we apply a novel text-mining method that uses a zigzagged search that reinvestigates past documents based on the updated chain. We also utilize social networks of news actors to reveal connections among news articles. We conduct 2 user studies in terms of 4 effectiveness measures—*relevance*, *coverage*, *coherence*, and *ability to disclose relations*. The first user study compares several versions of the framework, by varying parameters, to set a guideline for use. The second compares the framework with 3 baselines. The results show that our method provides statistically significant improvement in effectiveness in 61% of pairwise comparisons, with medium or large effect size; in the remainder, none of the baselines significantly outperforms our method.

Introduction

A story chain is a set of related text documents, each with a different event. In our case, a story chain is constructed for a given document in a news collection. We use the phrases "story chain" and "news chain" interchangeably. Discovering news chains i) reveals how events are connected and, thus, enables users to easily understand the big picture of events; ii) makes news consumers become aware of hidden relations among events; iii) detects different aspects of the input story; and iv) helps avoid information overload. Some of the possible application domains of story-chain discovery are investigative journalism, in which journalists or researchers examine a specific news topic; the analysis of intelligence reports (Hossain, Butler, Boedihardjo, & Ramakrishnan, 2012), patents (Tseng, Lin, & Lin, 2007), and legal documents (Stranieri & Zeleznikow, 2011).

A good story chain has a set of properties. Shahaf and Guestrin (2012) argue that *relevance* between input and chain members should be high. *Coherence* is another important property of news chains, which means a low relevance gap in the transition between any two chain members. Zhu and Oates (2014) expand the characteristics of news chains with measures of low *redundancy* and high *coverage*. A story chain has low redundancy when it includes no more than one representative for each event, and high coverage when it covers different aspects of the story. In addition to these, we also consider if previously unknown relations among news actors are disclosed by the chain—we call it the *ability to disclose relations*. Figure 1 shows a sample story chain with five documents.

We present a framework for story-chain discovery based on three complementary parts that i) scan the collection, ii) measure the similarity between chain-member candidates and the chain, and iii) measure similarity among news articles by exploiting lexical features and news actors. We discover story chains with a novel approach that uses a sliding-time window that updates the current chain incrementally. Inside the window, for the first time in this domain, we introduce zigzagged search that reinvestigates past documents based on the updated chain. Zigzagged search imitates the forward-and-backward search behavior of an investigative journalist.

Received May 13, 2016; revised January 13, 2017; accepted March 8, 2017

© 2017 ASIS&T • Published online 2 September 2017 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/asi.23885

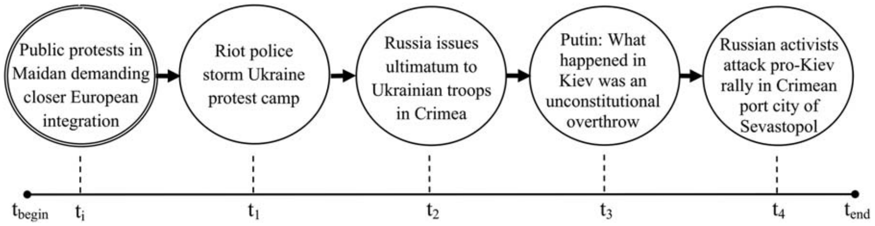

Public protests in Maidan demanding closer European integration
Riot police storm Ukraine protest camp
Russia issues ultimatum to Ukrainian troops in Crimea
Putin: What happened in Kiev was an unconstitutional overthrow
Russian activists attack pro-Kiev rally in Crimean port city of Sevastopol
t_{begin}
t_i
t_1
t_2
t_3
t_4
t_{end}

FIG. 1. A sample story chain with five documents that tells a story that connects public protests in Ukraine with Russian independence activists in Crimea. The input document is double circled with timestamp t_i. The beginning and end of the collection are t_{begin} and t_{end}, respectively.

The contributions of this study are the following. We

a. Develop a story-chain discovery framework that employs zigzagged search and news actors.
b. Conduct two user studies:
   1. The first finds a guideline for using the framework by answering the following research questions:
      - What is the proper time-window length to be used while scanning the collection?
      - How should we measure the similarity between a news chain and a candidate article?
      - When a social network of news actors is utilized, is it necessary to use a large network of news actors instead of exploiting a subset of important actors?
      - Which similarity method performs better in news-chain discovery: lexical features using the vector space model, or meta features based on news actors? Can we improve the effectiveness by using multiple methods together in a hybrid approach?
   2. The second compares our method with baselines to answer:
      - What are the benefits of our framework against baseline approaches?
c. Support user studies with statistical tests, which can set an example for similar studies.
d. Integrate our framework into a real-time news aggregator to observe its practical implications.

In the next section, we summarize the related work for story-chain discovery. We then explain the details of our framework, present the user studies and their results, and finally conclude the paper with a summary and some future research pointers.

Related Work

Simple Story Chains

In Topic Detection and Tracking (TDT) (Allan, 2002; Can et al., 2010), a topic is defined as an event or activity, with all directly related events. Since news articles are related to the same topic, we refer to such chains as simple story chains. TDT has a task called *link detection* that "detects whether a pair of stories discuss the same topic." In our case, the purpose of story chains is not restricted to detect relations in the same topic, but also coherent connections among different topics.

Cluster-Based Story Chains

Mei and Zhai (2005) cluster similar documents to obtain trends or themes in time, and then, clusters are connected to exhibit evolutionary theme patterns. Subasic and Berendt (2010) examine evolutionary theme patterns using interactive graphs. Nallapati, Feng, Peng, and Allan (2004) introduce event threading in clusters of events, and find dependencies among these clusters in a tree structure. Yang, Shi, and Wei (2009) develop event evolution graphs, which present underlying structure and relations among events of a topic. Kim and Oh (2011) apply topic modeling to uncover groups that contain related documents; chains are then constructed by finding similar topics in a time window. Song et al. (2016) develop a topic modeling approach to model documents and concept drifts in a tree structure. Shahaf, Guestrin, Horvitz, and Leskovec (2015) connect sets of clusters of news articles in a timeline to cover different aspects of the same topic. They find overlaps among clusters of different chains to reveal the evolution of the story.

Complex Story Chains

We define that a story chain is complex if it reveals relations among events of different topics. Complex story chains were first studied by Shahaf and Guestrin (2012). Giving two input news articles, their aim was to find a coherent story chain that connects them by maximizing the influence of the weakest connection. Influence is a measure to find similarity between two documents using the random walk theory. Zhu and Oates (2014) claim to improve the approach of Shahaf and Guestrin (2012) in terms of efficiency and redundancy. They use the inner structure of news articles by extracting named entities.

The main differences of our study from others are the following:

1. We introduce zigzagged search to discover story chains.
2. We exploit social networks of news actors to reveal connections among news articles.
3. Our user studies are supported by statistical tests.

4. We integrate our method into a real-time news aggregator to observe practical issues.
5. Our input is only a news article that indicates the start of a chain, instead of taking both start and end, that is, connecting two dots (Shahaf & Guestrin, 2012) (they do not develop an algorithm from scratch, but adapt their two-input algorithm to the one-input problem by extending it with user's feedback).

Other Related Studies

Timeline summarization methods give a summary of a query event with timeline projection (Yan et al., 2011). Some studies create a hierarchy in the timeline for the given text collection (Kleinberg, 2003). Given two objects, storytelling studies (Kumar, Ramakrishnan, Helm, & Potts, 2008) aim to explicitly relate them by using their intersections; for instance, two text documents (abstracts) are linked by finding word intersections. Choudhary, Mehta, Bagchi, and Balakrishnan (2008) find actors and their interactions in a given news collection. Similar studies support intelligence analysts to suggest unknown relations among entities (Hossain et al., 2012).

A Framework for Story-Chain Discovery

In this section, we present a temporal text-mining framework for story-chain discovery. Figure 2 shows an illustration of the framework that includes three complementary parts: 1) A given collection is scanned by using a sliding-time window that uses zigzagged search. Assume that the current chain includes four documents, labeled w, x, y, z; the first candidate document to be added to the chain is labeled a. 2) Documents are added to the chain according to the similarity between the candidate and the chain. 3) Similarity between two documents is measured by employing a social network of news actors to reveal connections among news articles. We also calculate similarities based on the vector space, named-entity, and hybrid model.

Scanning the Collection

We scan a given collection to search news articles, related to the input, by using a sliding-time window that uses zigzagged search; Figure 3 shows an example. The timeline is divided into nonoverlapping windows with a fixed-length (w) in days. We use a time window-based approach to update the current chain incrementally by considering only the members of the window. The user selects a news article d_i with the timestamp t_i, where $1 \leq \ i \ \leq N$, and N is the number of documents in the collection. An initial news chain is created with d_i. The first window is defined for $[t_i, \quad t_i+w)$. The time window is not allowed to exceed the ending time of the collection. If the similarity between d_i and d_c, a candidate news article inside the window, is higher than a threshold value θ, then d_c is added to the news chain.

We propose a forward-and-backward zigzagged search. We expect that making a zigzag in the timeline reveals missed news articles by using newly added documents to the chain. To do so, after processing the last news article in terms of time in the window, a new search phase on the same window is started by going backwards in the timeline. All news articles until the beginning of the window are processed in this backward-search phase. However, in this phase, similarity is calculated between the current chain, which is updated in the forward phase, and the candidate document. After a zigzag is completed, the same process is repeated by sliding the window by w days, until the remaining news articles are processed.

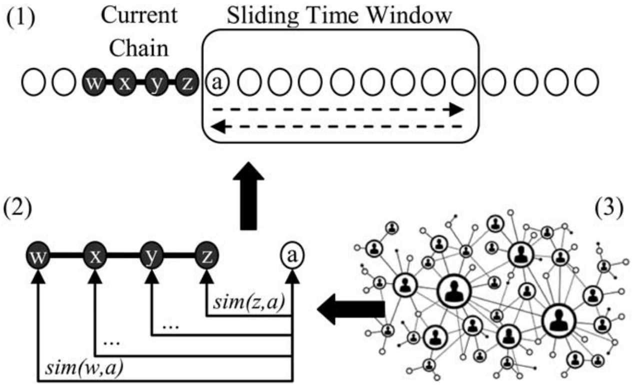


FIG. 2. An illustration of the proposed framework that uses a sliding-time window with zigzagged search, and a social network of news actors.

Similarity of Candidate With News Chain

While processing each candidate document to be added to the chain, we measure its similarity with the chain, which is represented by all of its current members. We also assign weights to similarity scores between the candidate and chain members. We call these methods *all members* and *weighted members*, respectively.

All members. Similarity scores between a candidate document, d_c, and chain members are measured as follows, where h is the current chain.

$$sim_{all}(d_c, h)=\left(\sum_{i\in h} sim(d_i,\ d_c)\right)/|h| \tag{1}$$

Weighted members. We assign weights to $sim(d_c, h)$, according to the closeness of the candidate document to the chain as follows, where $w_i{=}r_i/|h|$, and r_i is the order of the document, d_i, in the chain. We expect to improve the coherence of the chain. For simplicity, it is assumed that w_i is calculated to add a candidate document to the end of the chain.

$$sim_{weighted}(d_c, h)=\left(\sum_{i\in h} sim(d_i,\ d_c)\ \times w_i\right)/|h| \tag{2}$$

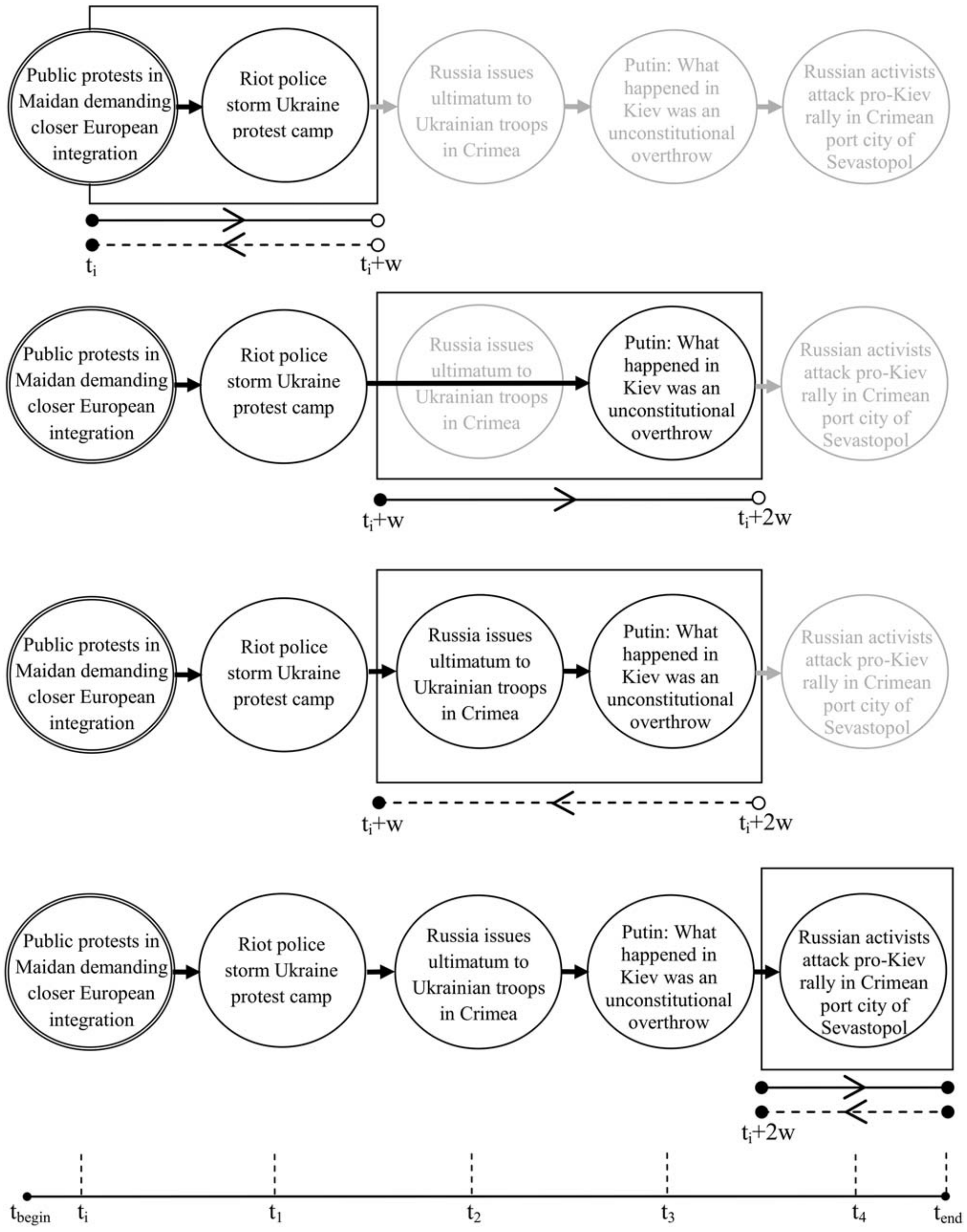

Public protests in Maidan demanding closer European integration
Riot police storm Ukraine protest camp
Russia issues ultimatum to Ukrainian troops in Crimea
Putin: What happened in Kiev was an unconstitutional overthrow
Russian activists attack pro-Kiev rally in Crimean port city of Sevastopol
t_i
t_i+w
t_i+2w
t_{begin}
t_1
t_2
t_3
t_4
t_{end}

FIG. 3. Discovery of a sample story chain by using a sliding-time window with zigzagged search. The beginning and end of the collection is t_{begin} and t_{end}, respectively. Window length is w days. The beginning of a window is inclusive, and the end is exclusive, shown by filled-in and empty circles, respectively. The input is the double-circled news article that has the timestamp t_i, mentioning the beginning of public protests in Ukraine in 2014. After three windows are processed from t_i to t_{end}, the bottommost chain is the output chain with five documents telling a story that connects public protests in Ukraine with Russian independence activists in Crimea. This chain is an extracted version of the output of the *hZZ*—namely, the hybrid algorithm, to be defined later, of the *Ukrainian Riots* case.

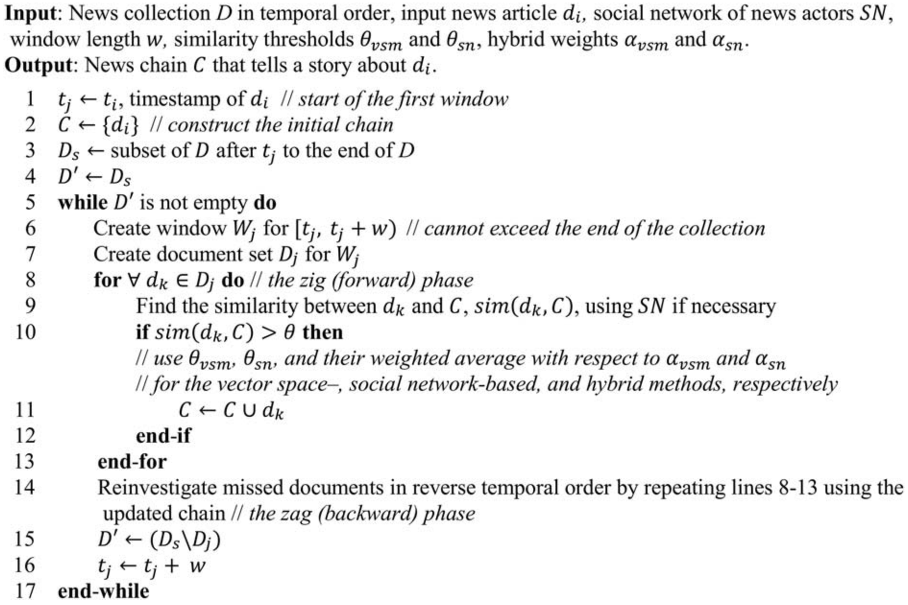
Similarity Between News Articles

We propose four methods for measuring the similarity between two documents.

Vector space-based similarity. In the vector space model, documents are represented with word vectors that are sets of unique tokens in the collection. Each word is assigned to a weight by using term frequency. We calculate similarity between two document vectors by the *cosine similarity* measure. We use a stop word list—an extended version of the list given in Can et al. (2008)—and *F5* stemming, which uses the first five letters of each word, and shows good performance in information retrieval (Can et al., 2008) and news categorization (Toraman, Can, & Koçberber, 2011). We use the phrases *vector space model* and *cosine similarity* interchangeably.

Named entity-based similarity. Named entity recognition (NER) is the task of information extraction to identify and classify important elements in a text document (Nadeau & Sekine, 2007). In this study, named entities are detected for people, organizations, and locations. We employ the named-

DOI: 10.1002/asi



```
Input: News collection $D$ in temporal order, input news article $d_i$, social network of news actors $SN$,
 window length $w$, similarity thresholds $\theta_{vsm}$ and $\theta_{sn}$, hybrid weights $\alpha_{vsm}$ and $\alpha_{sn}$.
Output: News chain $C$ that tells a story about $d_i$.
 1  $t_j \leftarrow t_i$, timestamp of $d_i$ // start of the first window
 2  $C \leftarrow \{d_i\}$ // construct the initial chain
 3  $D_s \leftarrow$ subset of $D$ after $t_j$ to the end of $D$
 4  $D' \leftarrow D_s$
 5  while $D'$ is not empty do
 6      Create window $W_j$ for $[t_j,\ t_j + w)$ // cannot exceed the end of the collection
 7      Create document set $D_j$ for $W_j$
 8      for $\forall\ d_k \in D_j$ do // the zig (forward) phase
 9          Find the similarity between $d_k$ and $C$, $sim(d_k, C)$, using $SN$ if necessary
10          if $sim(d_k, C) > \theta$ then
            // use $\theta_{vsm}$, $\theta_{sn}$, and their weighted average with respect to $\alpha_{vsm}$ and $\alpha_{sn}$
            // for the vector space–, social network-based, and hybrid methods, respectively
11              $C \leftarrow C \cup d_k$
12          end-if
13      end-for
14      Reinvestigate missed documents in reverse temporal order by repeating lines 8-13 using the
        updated chain // the zag (backward) phase
15      $D' \leftarrow (D_s \backslash D_j)$
16      $t_j \leftarrow t_j + w$
17  end-while
```

FIG. 4. The framework algorithm for story chain-discovery.

entity-recognition program of Küçük and Yazıcı (2011) that uses a rule-based approach. However, its output is noisy; since multiple entities refer to the same object, for example, synonyms, and there are several entities with missing parts. Such problems can be solved by named-entity resolution (Cucerzan, 2007). We obtain all named entities in the collection, and apply heuristic rules that check if named entities can be resolved by their preceding or succeeding words. We also merge synonyms of popular objects into the same entity.

After named entities are determined, the similarity between two news articles, d_i and d_j, is measured by the Dice similarity coefficient, as follows, where N_c is the number of common unique actors in d_i and d_j; N_i and N_j are the number of unique actors in d_i and d_j, respectively.

$$sim_{namedEntity}\left(d_i, d_j\right) = 2 \times N_c / \left(N_i + N_j\right) \tag{3}$$

Social network-based similarity. Social network studies aim to reveal relations among social actors in a network structure (Liu, 2011). We create a social network of news actors (named entities) for the entire collection, where edges represent relations. We detect news actors as described above, and create an edge between two actors if both occur in the same document. We use the Dice coefficient for assigning weights to edges. The edge weight, $w(a,b)$, between two actors, a and b, in a social network is determined as follows, where N_c is the number of documents in which both actors occur, N_a and N_b are the numbers of documents that include the actors a and b.

$$w(a,b) = 2 \times N_c / (N_a + N_b) \tag{4}$$

The similarity between two documents, d_i and d_j, is then measured as follows, where A_i and A_j are the sets of unique actors in d_i and d_j, and N_p is the number of all unique pairs between the actors of d_i and d_j.

$$sim_{socialNetwork}\left(d_i, d_j\right) = \left(\sum_{a \in A_i} \sum_{b \in A_j} w(a,b)\right) / N_p \tag{5}$$

The difference between the named entity and social network-based similarity methods is that the former considers only the co-occurrence of actors between two news articles; the latter uses edge weights in a social network, that is, relations among actors of two news articles.

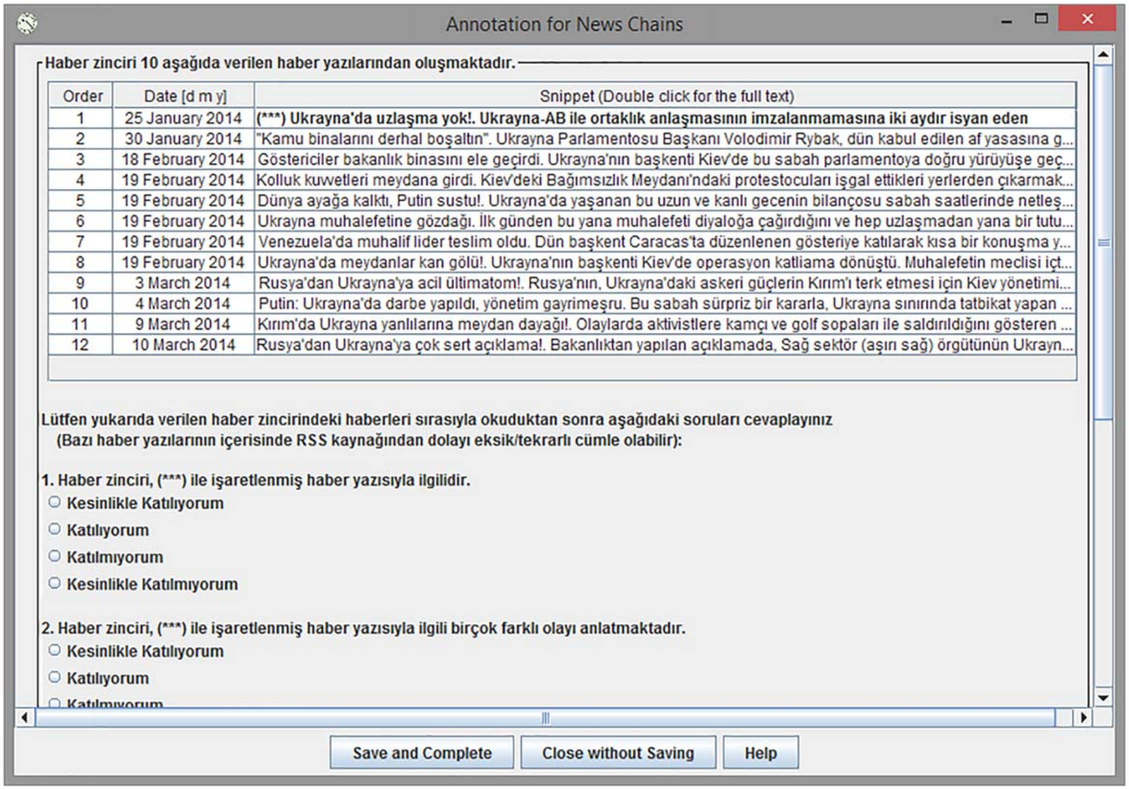
Hybrid similarity. The hybrid similarity between two documents, d_i and d_j, is a linear combination of the similarity scores of n methods:

$$sim_{hybrid}\left(d_i, d_j\right) = \sum_{k=1}^{n} sim_k\left(d_i, d_j\right) \times \alpha_k \tag{6}$$

Each method k outputs a score for the similarity between d_i and d_j as $sim_k\left(d_i, d_j\right)$; however, there is a need for the calibration of different methods. The parameter α_k is a significance coefficient for the method k ($0 \leq \alpha_k \leq 1$, $\sum_{k=1}^{n} \alpha_k = 1$). We combine lexical features, namely, the vector space model, and social network, in the hybrid model by setting α values equal to 0.5.

The Framework Algorithm

The framework algorithm for story chain-discovery is given in Figure 4.



Haber zinciri 10 aşağıda verilen haber yazılarından oluşmaktadır.

Order	Date [d m y]	Snippet (Double click for the full text)
1	25 January 2014	**(***) Ukrayna'da uzlaşma yok!. Ukrayna-AB ile ortaklık anlaşmasının imzalanmamasına iki aydır isyan eden**
2	30 January 2014	"Kamu binalarını derhal boşaltın". Ukrayna Parlamentosu Başkanı Volodimir Rybak, dün kabul edilen af yasasına g...
3	18 February 2014	Göstericiler bakanlık binasını ele geçirdi. Ukrayna'nın başkenti Kiev'de bu sabah parlamentoya doğru yürüyüşe geç...
4	19 February 2014	Kolluk kuvvetleri meydana girdi. Kiev'deki Bağımsızlık Meydanı'ndaki protestocuları işgal ettikleri yerlerden çıkarmak...
5	19 February 2014	Dünya ayağa kalktı, Putin sustu!. Ukrayna'da yaşanan bu uzun ve kanlı gecenin bilançosu sabah saatlerinde netleş...
6	19 February 2014	Ukrayna muhalefetine gözdağı. İlk günden bu yana muhalefeti diyaloğa çağırdığını ve hep uzlaşmadan yana bir tutu...
7	19 February 2014	Venezuela'da muhalif lider teslim oldu. Dün başkent Caracas'ta düzenlenen gösteriye katılarak kısa bir konuşma y...
8	19 February 2014	Ukrayna'da meydanlar kan gölü!. Ukrayna'nın başkenti Kiev'de operasyon katliama dönüştü. Muhalefetin meclisi içt...
9	3 March 2014	Rusya'dan Ukrayna'ya acil ültimatom!. Rusya'nın, Ukrayna'daki askeri güçlerin Kırım'ı terk etmesi için Kiev yönetimi...
10	4 March 2014	Putin: Ukrayna'da darbe yapıldı, yönetim gayrimeşru. Bu sabah sürpriz bir kararla, Ukrayna sınırında tatbikat yapan ...
11	9 March 2014	Kırım'da Ukrayna yanlılarına meydan dayağı!. Olaylarda aktivistlere kamçı ve golf sopaları ile saldırıldığını gösteren ...
12	10 March 2014	Rusya'dan Ukrayna'ya çok sert açıklama!. Bakanlıktan yapılan açıklamada, Sağ sektör (aşırı sağ) örgütünün Ukrayn...

Lütfen yukarıda verilen haber zincirindeki haberleri sırasıyla okuduktan sonra aşağıdaki soruları cevaplayınız
(Bazı haber yazılarının içerisinde RSS kaynağından dolayı eksik/tekrarlı cümle olabilir):

1. Haber zinciri, (***) ile işaretlenmiş haber yazısıyla ilgilidir.
○ Kesinlikle Katılıyorum
○ Katılıyorum
○ Katılmıyorum
○ Kesinlikle Katılmıyorum

2. Haber zinciri, (***) ile işaretlenmiş haber yazısıyla ilgili birçok farklı olayı anlatmaktadır.
○ Kesinlikle Katılıyorum
○ Katılıyorum
○ Katılmıyorum

FIG. 5. A sample screen from the annotation program. The current chain to be annotated, regarding *Ukrainian Riots*, is given at the top of screen. Annotators have to read all news articles in order, and then answer all questions. [Color figure can be viewed at wileyonlinelibrary.com]

User Studies

To the best of our knowledge, there is no ground truth for the evaluation of story-chain discovery algorithms. For this reason, we conducted two user studies.[1] The first compares several versions of the framework, by varying parameters, to set a guideline for use. The second compares the framework with three baseline methods.

User Study Setup

Collection. Chains were discovered in a news collection that included 1,656 documents from the *Sözcü* newspaper (http://www.sozcu.com.tr) between December 20, 2013, and March 11, 2014. The number of detected named entities was 4,957, of which 2,890 were people, 915 were organizations, and 1,152 were locations.

Three news cases (topics) were used in our user studies. The first case is the riots and protests against the Ukrainian government, demanding closer European integration, which started in November 2013, and is referred to as *Ukrainian Riots* in this study. The second case is the trucks that were pulled over while going from Turkey to Syria by the military police, claiming that they carried illegal ammunition, in January 2014, referred to as *Trucks Going to Syria*. The last case is the domestic match-fixing allegations of the Fenerbahçe football team, started in July 2011, referred to as *Allegations of Fenerbahçe*. We selected three input news articles representing the cases. The dates of the input documents for each topic are January 25, 2, and 17 of 2014, respectively.

Annotation program. User studies were conducted on an annotation program. Annotators were assigned the same tasks. A sample screenshot for the annotation screen is given in Figure 5. The chain members are listed chronologically. The full text is visible in a pop-up window if a news article is double-clicked. The snippet of input is displayed in bold with three consecutive stars. At the bottom of the chain, questions are asked to assess the chain quality.

Evaluation measures. In similar studies, Shahaf and Guestrin (2012) evaluated story chains according to relevance, coherence, and redundancy. Zhu and Oates (2014) consider coverage, in addition to other measures. We also assess if previously unknown relations among news actors are disclosed by the chain—ability to disclose relations.

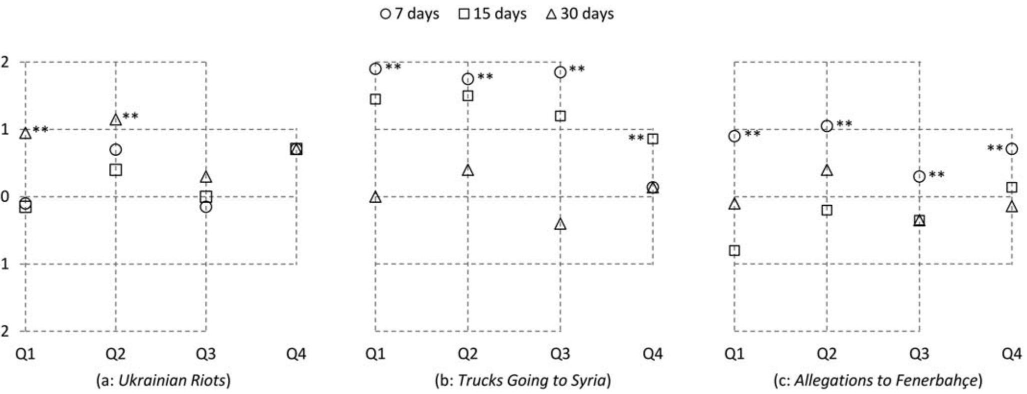
We gave annotators five statements and asked them to label to what extent they agree that i) the news article is relevant to the input document marked with (***); ii) the news chain covers different events related to the input; iii) there are no redundant documents in the chain; iv) the chain is

[1] The materials that we are unable to give due to the limited space are provided in the details webpage (http://cs.bilkent.edu.tr/~ctoraman/story_chains); such as the text collection, output story chains, annotations, and details of statistical tests.

TABLE 1. The design of user study 1: Eight versions (A to H) of the framework algorithm.

(Decision no.) Research question	Version	Similarity of candidate with news chain	Similarity between news articles	Window length in days
(Decision 1)	A	ALL	VSM	7
Is there any proper window length?	B	ALL	VSM	15
	C	ALL	VSM	30
(Decision 2)	B	ALL	VSM	15
Which method for candidate similarity?	D	Weighted members	VSM	15
(Decision 3)	E	ALL	SN (all actors)	15
What type of social network?	F	ALL	SN (top 500 actors)	15
(Decision 4)	B	ALL	VSM	15
Which method for document similarity?	E	ALL	SN (all actors)	15
	G	ALL	Named entity	15
	H	ALL	Hybrid (SN & VSM)	15

Note. ALL: the all-members method, SN: social network, VSM: the vector space model.



(a: *Ukrainian Riots*) (b: *Trucks Going to Syria*) (c: *Allegations to Fenerbahçe*)

FIG. 6. Annotation results for Decision 1: *proper window length*. Subfigures (a, b, and c) are for the results of three topics. Question numbers are given in horizontal axis (Q1: *Relevance*, Q2: *Coverage*, Q3: *Coherence*, and Q4: *Ability to disclose relations*). The vertical axis represents an average score of annotation answers (scale is between −2 and 2 for Q1–Q3, −1 and 1 for Q4). For the pairwise comparison of the top two algorithms, "**" means that there is a statistically significant increase at the 1% level ($p < .01$), after the corresponding method is applied. See Table 2 for details of statistical tests. The same notation is used in the following figures.

coherent, that is, two adjacent documents are on the same topic (if they are not on the same topic, they are still related within the context of the input); and v) after reading the chain, new relations among news actors (people, organizations, and places) are learned.

All questions have text answers that are given in *positiveness* order, which are mapped to an integer scale of 2, 1, −1, and −2. The average of all annotators is taken for each question. The neutral choice of zero is not given, to make annotators think more critically, and prevent selecting the first alternative choice that has the minimum cognitive requirements (Krosnick et al., 2002). The last question has two answers, for having the ability to disclose relations or not, mapped to 1 and −1.

Annotators. All tasks were assigned to 20 annotators in the first and 12 in the second user study. Annotators were mostly graduate students, and a few undergraduates and faculty members. For the first user study, Fleiss's kappa (1971) score is 0.49 for relevance, 0.46 for coverage, 0.12 for redundancy, 0.30 for coherence, and 0.33 for ability to disclose relations. For the second, the same scores were 0.63, 0.34, 0.02, 0.23, and 0.18. Since both redundancy scores are below 0.20, meaning slight agreement among annotators, according to the interpretation of Landis and Koch (1977), we ignore the results of redundancy.

User Study 1: Varying Framework Parameters

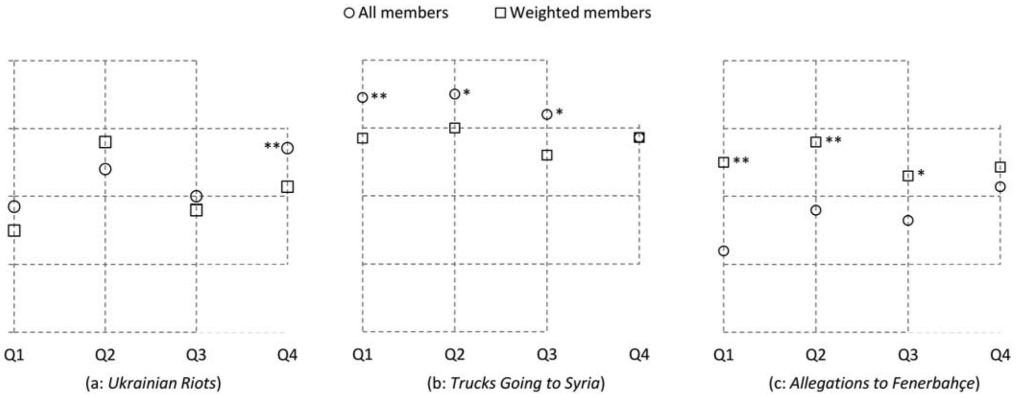
Methodology. The first user study consisted of 24 chains, obtained by the framework algorithm, with eight sets of parameters (versions A to H) on three topics. The design of this user study, given in Table 1, is based on the first four research questions asked in the Introduction.

In the decisions, we compared the performance of two or more versions to answer their respective questions. Decisions were independent of each other, that is, a decision

TABLE 2. The details of the Friedman test for Decision 1 with respect to Figure 6.

	Ukrainian Riots			*Trucks Going to Syria*			*Allegations of Fenerbahçe*		
Q: *measure*	*p*	Chi	Pw. d	*p*	Chi	Pw. d	*p*	Chi	Pw. d
Q1: *relevance*	<.01	16.93	1.05	<.01	29.66	0.55	<.01	23.35	1.00
Q2: *coverage*	<.01	9.46	0.45	<.01	28.17	0.25	<.01	19.73	0.65
Q3: *coherence*	—	—	0.30	<.01	29.38	0.65	0.012	8.70	0.65
Q4: *disclose relations*	—	—	—	0.011	8.93	0.72	<.01	14.71	0.57

Note. The Friedman test's *p*-values are listed with chi-square values. "Pw. d" is the mean difference between the top two algorithms. The *p*-values of the pairwise comparisons of the top two algorithms are marked in Figure 6. The same notation is used in the following tables.



(a: *Ukrainian Riots*)

(b: *Trucks Going to Syria*)

(c: *Allegations to Fenerbahçe*)

FIG. 7. Annotation results for Decision 2: *all members* vs. *weighted members*. Note that "*" means that there is a statistically significant increase at the 5% level ($p < .05$), after the corresponding method is applied. See Table 3 for details of statistical tests. The same notation is used in the following figures.

result is not used in later decisions. For a fair evaluation, all parameters, except the one we want to gauge its effect on the algorithm, are kept the same. Based on the observations in preliminary experiments, fixed parameters were selected as all members, vector space model, and a window length of 15 days.

In preliminary experiments, we observed that long chains are overwhelming to comprehend. In both user studies, we used a heuristic approach that searches for effective similarity-threshold values (θ) in a greedy fashion, by incrementing with a constant value. For the sake of simplicity, it keeps the chain lengths to 15 or fewer documents. The chain length decreases as the threshold values increase. For instance, in the hybrid algorithm we reduce the chain length from 23 to 15 by incrementing the cosine and social-network thresholds from 0.155 to 0.160, and from 0.115 to 0.120, respectively. On average, there are 12 news articles in a chain (median: 13, minimum: 4, maximum: 15).

Results of user study 1. Decision 1: Is there any proper time-window length? While scanning with a zigzagged search, we employed a sliding window that captures news articles. In Figure 6, we examine three window lengths of 7, 15, and 30 days.

We observed that the performance of varying the window length is case-dependent. For cases with a uniformly distributed number of documents (*Trucks Going to Syria* and *Allegations of Fenerbahçe*), the window should be small—7 days in both cases—in order to not miss news articles in a dense collection. For a nonuniformly distributed number of documents (*Ukrainian Riots*), the window should be large (30 days), to catch news articles in a sparse collection.

In order to test whether a case is uniformly distributed, we applied the Shapiro–Wilks test (1965) that states that, with small *p*-values, the collection does not follow a uniform distribution. In order to apply the test, we divided the collection into intervals of 20 days, and counted the number of articles for each case. For *Trucks Going to Syria*, *Allegations of Fenerbahçe*, and *Ukrainian Riots*, *p*-values are .30, .50, and .10, respectively; *Ukrainian Riots* seems to be less uniformly distributed than *Trucks Going to Syria* and *Allegations of Fenerbahçe*.

The Friedman test (1937) was applied to the results of Figure 6; the details are given in Table 2. The Friedman test shows if there is a significant difference between at least two methods. This test is applied when there are more than two methods (groups), annotator answers are ordinal-categorical, and annotations (observations) are paired and nonuniformly distributed. We used the one-tailed *p*-values instead of the two-tailed, since we try to show that the effectiveness of one algorithm is greater than the effectiveness of the others, instead of being equal. All statistical tests in this study were conducted in the same manner.

In order to have pairwise comparisons, we further applied the post-hoc test proposed by Conover (1999), which is valid if the Friedman test indicates any significance. In Figures 6–9,

DOI: 10.1002/asi

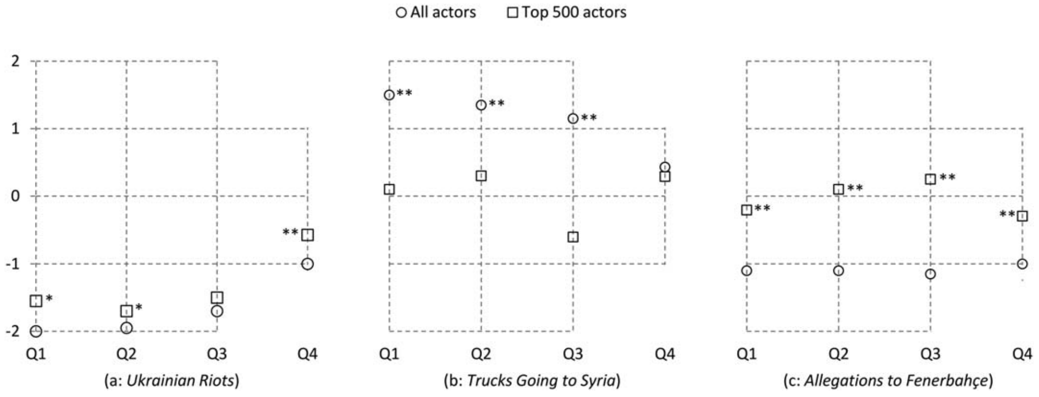


FIG. 8. Annotation results for Decision 3: *All news actors* vs. *top 500 news actors*. See Table 4 for details of statistical tests.

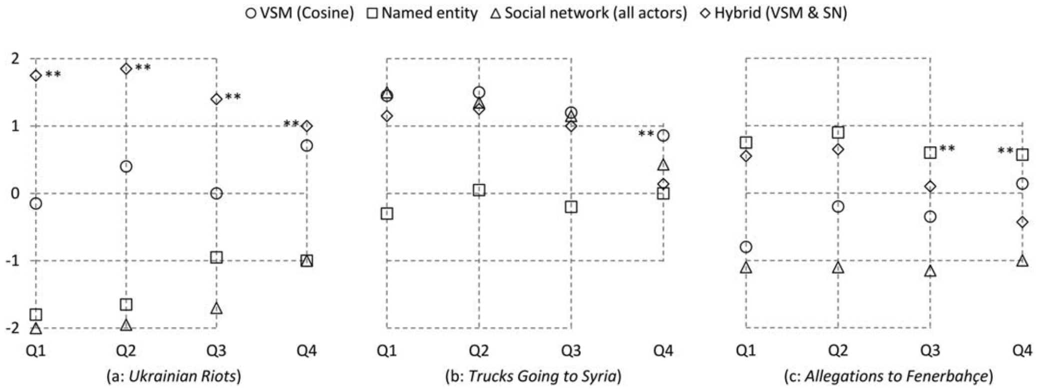


FIG. 9. Annotation results for Decision 4: *similarity methods*. See Table 5 for details of statistical tests.

the Conover test results are given for only the top two algorithms, since we want to see the significance of the winner. The scores of the Conover tests are provided in the details webpage.

Decision 2: Which method for candidate similarity works better? The effectiveness of the all-members and weighted-members methods depends on the *freshness* of the input, as depicted in Figure 7. An input is fresh if it is close to the beginning of the topic. Note that the input documents of all cases are from January 2014. The *Trucks Going to Syria* event starts in January 2014, and *Ukrainian Riots* in November 2013. *Allegations of Fenerbahçe* is relatively old, beginning in July 2011; the weighted-members method works better for this case, since it gives lower importance to old members of the chain, including the input that is not fresh. For other cases where we have relatively fresher inputs, the all-members method is more effective in terms of relevance, coherence, and ability to disclose relations, since it gives the same importance to all members of the chain, including the fresh input. Our expectation of weighted members providing more coherent chains fails in some cases.

The Wilcoxon signed-ranks test (1945) was applied to the results of Figure 7 to see any significant difference between algorithms; the details are given in Table 3. This test is used when there are two methods, annotator answers are ordinal-categorical, and annotations are paired and non-uniformly distributed.

Decision 3: What size of social network works better? The results of using all news actors and the top 500 most important ones, in terms of frequency, are given in Figure 8. We observe that using all news actors—∼5,000—is more effective for *Trucks Going to Syria*, which has more number of *minor* actors that are observed with less frequency in the collection. For two cases with more number of *major* actors (*Allegations of Fenerbahçe* and *Ukrainian Riots*), using the top 500 important news actors is more effective. Using all news actors for such cases reduces the effectiveness scores, due to possible inclusion of redundant ones. The Wilcoxon test was applied to the results of Figure 8; the details are given in Table 4.

Decision 4: Which method for document similarity works better? The results of four similarity methods are given in Figure 9. The success of the hybrid model, which employs both lexical features and news actors, is case-dependent. For the cases with a relatively higher number of major actors (*Ukrainian Riots* and *Allegations of Fenerbahçe*), the effectiveness of the vector space model is increased by the hybrid model—the only exception is ability to disclose relations of *Allegations of Fenerbahçe*. For the case with a relatively higher number of minor actors (*Trucks Going to Syria*), the effectiveness of the vector space model is not increased by employing news actors.

TABLE 3. The details of the Wilcoxon test for Decision 2 with respect to Figure 7, where "d" is mean difference, "p," "Z," and "r" are scores of the Wilcoxon test.

	Ukrainian Riots				Trucks Going to Syria				Allegations of Fenerbahçe			
Q: measure	d	p	Z	r	d	p	Z	r	d	p	Z	r
Q1: *relevance*	0.35	0.074	−2.46	−0.93	0.60	0.003	−2.75	−0.92	1.30	<.001	−3.72	−1.03
Q2: *coverage*	0.40	—	—	—	0.50	0.011	−2.29	−0.66	1.00	<.001	−3.12	−0.99
Q3: *coherence*	0.20	—	—	—	0.60	0.033	−1.88	−0.54	0.65	0.020	−2.05	−0.68
Q4: *disclose relations*	0.57	0.003	−2.75	−0.97	—	—	—	—	0.29	—	—	—

Note. The same notation is used in the following similar tables when the Wilcoxon test is applied.

TABLE 4. The details of the Wilcoxon test for Decision 3 with respect to Figure 8.

	Ukrainian Riots				*Trucks Going to Syria*				*Allegations of Fenerbahçe*			
Q: *measure*	d	p	Z	r	d	p	Z	r	d	p	Z	r
Q1: *relevance*	0.45	0.013	−2.23	−0.91	1.40	<.001	−3.29	−0.88	0.90	0.002	−2.88	−0.91
Q2: *coverage*	0.25	0.036	−1.80	−0.68	1.05	0.004	−2.65	−0.73	1.20	<.001	−3.29	−0.91
Q3: *coherence*	0.20	—	—	—	1.75	<.001	−3.29	−0.75	1.40	<.001	−3.19	−0.85
Q4: *disclose relations*	0.43	0.009	−2.36	−0.96	0.14	—	—	—	0.71	0.005	−2.58	−0.97

Another observation is that using only named entities—as observed in the TDT domain (Can et al., 2010; Kumaran & Allan, 2004)—or only social networks performs poorly. However, the named entity method is more effective than the other methods, in the case of *Allegations of Fenerbahçe*. This can be explained by the fact that this case mostly involves the actor, *Aziz Yıldırım*, who is the club chairman, and not involved in any other case in the given collection. When the case involves many actors, as in *Ukrainian Riots* and *Trucks Going to Syria*, we observe that the effectiveness of using a social network, revealing relations among news actors, is higher than the effectiveness of using only named entities. The Friedman test was applied to the results of Figure 9; the details are given in Table 5.

Recommendations. Based on the results of the first user study, for parameter tuning, we recommend the use of:

a. *Dynamic window length*: When news articles are uniformly distributed, the window should be small. It should be large for nonuniformly distributed cases.
b. *Case-dependent candidate-similarity method*: The weighted-members method works better for inputs that are not fresh, while the all-members method is more effective with relatively fresher inputs.
c. *Variable social-network size*: For improving efficiency, the size of a social network can be relatively small for cases with a higher number of major actors.
d. *Case-dependent document-similarity method*: Lexical features based on the vector space model are more effective in measuring similarity for cases with minor news actors. When a small number of major actors are involved, the performance of news actor methods can be competitive with the vector space model. The effectiveness of the vector space model and news actors can be improved by combining them in a hybrid model.

User Study 2: Comparison With Baselines

Methodology. For comparison, we need to select a representative version of our framework algorithm. We can apply our fine-tuning recommendations on each topic; however, to provide a fair evaluation, we use the same version. Since our contribution is to employ news actors and zigzagged search for story-chain discovery, we chose among versions that employ news actors (named entity, social network, and hybrid). Since using only named entities or only social networks has poor performance, we compare the hybrid version with three baselines—referred to as *hZZ*: *Hybrid and Zigzagged Search*. The design of User Study 2 is given in Table 6.

The first baseline is a simple TDT (Allan, 2004) approach, which examines all documents once, and adds a document to the chain by measuring the cosine similarity with the seed, that is, input document. The second is an adaptive TDT (Allan, Papka, & Lavrenko, 1998) approach, which is similar to simple TDT, except that it employs a window to scan documents, updating the event description after processing each window. This method is similar to our framework, but without using zigzagged search and news actors. In both methods, chain lengths are set to 15 or fewer documents.

The third baseline is the search result list of Google News (http://news.google.com). The collection of Google News is a superset of our collection, since it includes *Sözcü* news. The title of the input document is given as a query string. For a fair comparison, we set the range of documents starting from the input date to the end date of our collection, and create a chain with the result list sorted in time. In *Allegations of Fenerbahçe*, since the list includes 40 documents (more than 15), we selected 11 equally spaced news articles.

Results of user study 2. The average scores are given in Table 7. In Table 8, the Friedman test was applied to show if there is a significant difference between at least two

TABLE 5. The details of the Friedman test for Decision 4 with respect to Figure 9.

	Ukrainian Riots			*Trucks Going to Syria*			*Allegations of Fenerbahçe*		
Q: *measure*	*p*	Chi	Pw. d	*p*	Chi	Pw. d	*p*	Chi	Pw. d
Q1: *relevance*	<.01	56.02	1.85	<.01	35.25	0.05	<.01	37.05	0.20
Q2: *coverage*	<.01	54.73	1.45	<.01	23.96	0.15	<.01	35.39	0.25
Q3: *coherence*	<.01	41.79	1.40	<.01	19.20	0.05	<.01	25.70	0.35
Q4: *disclose relations*	<.01	52.42	0.29	<.01	19.19	0.43	<.01	25.03	0.43

TABLE 6. The design of user study 2: Comparing our framework algorithm, *hZZ*: Hybrid and zigzagged search, with three baselines, *sTDT*: Simple TDT, *aTDT*: Adaptive TDT, *GN*: Google news.

Method name	Scanning the collection	Similarity of candidate with news chain	Similarity between news articles	Window length in days
sTDT	One pass with no window	Only with input document	VSM	—
aTDT	One pass with window	ALL	VSM	15
GN		Unknown		
hZZ	Zigzagged with window	ALL	Hybrid (SN & VSM)	15

TABLE 7. The average scores of all annotators for *sTDT*: Simple TDT, *aTDT*: Adaptive TDT, *GN*: Google news, *hZZ*: Hybrid and zigzagged.

	Ukrainian Riots				*Trucks Going to Syria*				*Allegations of Fenerbahçe*			
Q: *measure*	sTDT	aTDT	GN	hZZ	sTDT	aTDT	GN	hZZ	sTDT	aTDT	GN	hZZ
Q1: *relevance*	1.25	−1.25	0.33	**1.67**	−0.83	0.83	0.67	**1.08**	−1.50	−0.83	0.92	**1.00**
Q2: *coverage*	1.42	−0.33	0.83	**1.67**	−0.25	0.83	1.17	**1.25**	−1.00	−0.83	**0.67**	**0.67**
Q3: *coherence*	0.08	−1.25	0.08	**1.33**	−1.33	0.33	−0.58	**0.42**	−1.75	−0.83	−0.08	**0.00**
Q4: *disclose relations*	**0.83**	0.00	**0.83**	**0.83**	−0.17	0.50	**0.83**	**0.83**	−0.33	0.00	**0.83**	0.67

Note. The method(s) with the highest score is marked as bold.

methods. The methods are further pairwise compared with the Conover post-hoc test in Table 9. In order to measure the effect size of pairwise comparisons, we apply Cohen's d-test (1988). We highlight cells of Table 9 with dark gray if there is a large effect size, and light gray if medium; it remains white if it has a small effect size. The Cohen's d values and confidence intervals are provided in the details webpage.

In total, there were 72 pairwise comparisons between the methods. We have 36 pairwise comparisons in the rows of *hZZ*, which uses zigzagged search and a social network of news actors. The results show that it has statistically significantly higher relevance (67% of pairwise comparisons of *hZZ*), coverage (56% of pairs), coherence (78% of pairs), and ability to disclose relations (44% of pairs). We observe that our framework can be helpful to news consumers, since *hZZ* significantly improves the effectiveness with respect to baselines, in 61% of pairs (22 of 36 pairs); in the remainder, none of the baselines significantly outperforms our method. All of these pairs have medium (4 of 22 pairs) or large (18 of 22) effect sizes, according to the thresholds of Cohen (1992). Furthermore, we have medium effect sizes in two nonsignificant additional pairs.

TABLE 8. The details of the Friedman test with respect to Table 7.

	Ukrainian Riots		*Trucks Going to Syria*		*Allegations of Fenerbahçe*	
Q: *measure*	*p*	Chi	*p*	Chi	*p*	Chi
Q1: *relevance*	<.01	25.33	<.01	20.68	<.01	28.21
Q2: *coverage*	<.01	17.35	<.01	17.28	<.01	24.38
Q3: *coherence*	<.01	20.76	<.01	15.72	<.01	25.07
Q4: *disclose relations*	<.05	10.71	<.01	12.00	<.01	14.56

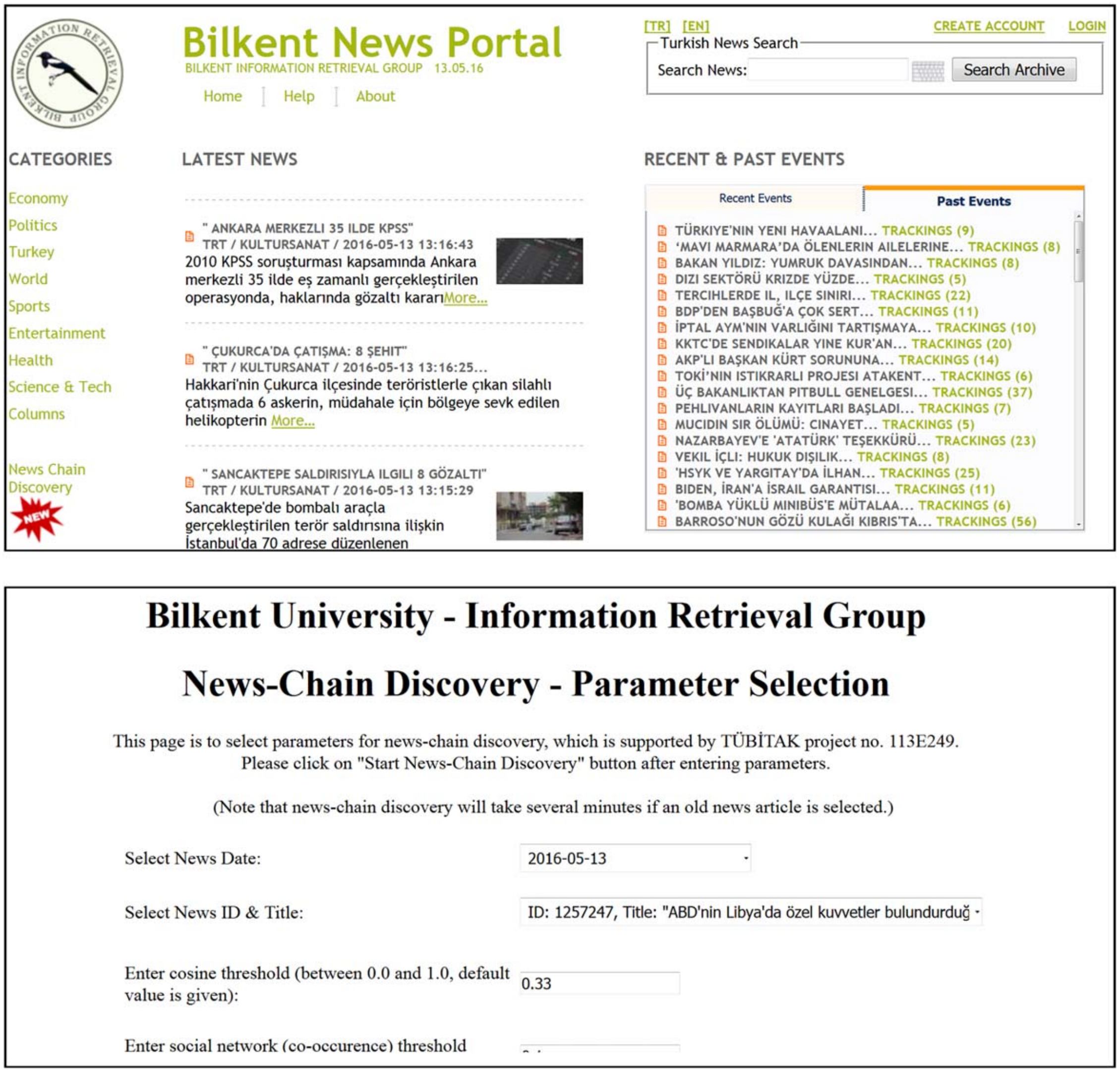
Practical Considerations

We employed the *hZZ* algorithm in the Bilkent News Portal (http://newsportal.bilkent.edu.tr), which aggregates Turkish news articles from various resources (Can et al., 2008). We integrated three social-network versions that include different numbers of news actors, by transforming them into matrices of news actors that involve edge weights. Sample screenshots of the system are given in Figure 10. The top screen is the front page of the portal, where the link of the news-chain discovery tool is provided in the left

TABLE 9. Pairwise comparisons of the methods in Table 7.

	Case	*Ukrainian Riots*			*Trucks Going to Syria*			*Allegations of Fenerbahçe*		
Q: measure	Method	sTDT	aTDT	GN	sTDT	aTDT	GN	sTDT	aTDT	GN
Q1: *relevance*	aTDT	−2.50**			+1.66**			+0.67**		
	GN	−0.92**	+1.58**		+1.50**	−0.16		+2.42**	+1.75**	
	hZZ	**+0.42***	**+2.92****	**+1.34****	**+1.91****	+0.25	+0.41	**+2.50****	**+1.83****	+0.08
Q2: *coverage*	aTDT	−1.75**			+1.08**			+0.17		
	GN	−0.59	+1.16**		+1.42**	+0.34		+1.67**	+1.50**	
	hZZ	+0.25	**+2.00****	**+0.84***	**+1.50****	+0.42	+0.08	**+1.67****	**+1.50****	0.00
Q3: *coherence*	aTDT	−1.33**			+1.66**			+0.92**		
	GN	0.00	+1.33**		+0.75**	−0.91**		+1.67**	+0.75**	
	hZZ	**+1.25****	**+2.58****	**+1.25****	**+1.75****	+0.09	**+1.00****	**+1.75****	**+0.83****	+0.08
Q4: *disclose relations*	aTDT	−0.83**			+0.67**			+0.33		
	GN	0.00	+0.83**		+1.00**	+0.33		+1.16**	+0.83**	
	hZZ	0.00	**+0.83****	0.00	**+1.00****	+0.33	0.00	**+1.00****	**+0.67****	−0.16

Note. Each cell includes the mean difference between the method scores, and the *p*-value of the Conover test if the difference is statistically significant. Note that "*" and "**" mean that there is a statistically significant increase at the 5% level ($p < .05$), and 1% level ($p < .01$), respectively. Large effect size is indicated with dark gray, and medium with light gray; small remains white.



" ANKARA MERKEZLI 35 ILDE KPSS"
TRT / KULTURSANAT / 2016-05-13 13:16:43
2010 KPSS soruşturması kapsamında Ankara merkezli 35 ilde eş zamanlı gerçekleştirilen operasyonda, haklarında gözaltı kararıMore...

" ÇUKURCA'DA ÇATIŞMA: 8 ŞEHIT"
TRT / KULTURSANAT / 2016-05-13 13:16:25...
Hakkari'nin Çukurca ilçesinde teröristlerle çıkan silahlı çatışmada 6 askerin, müdahale için bölgeye sevk edilen helikopterin More...

" SANCAKTEPE SALDIRISIYLA ILGILI 8 GÖZALTI"
TRT / KULTURSANAT / 2016-05-13 13:15:29
Sancaktepe'de bombalı araçla gerçekleştirilen terör saldırısına ilişkin İstanbul'da 70 adrese düzenlenen

TÜRKIYE'NIN YENI HAVAALANI... TRACKINGS (9)
'MAVI MARMARA'DA ÖLENLERIN AILELERINE... TRACKINGS (8)
BAKAN YILDIZ: YUMRUK DAVASINDAN... TRACKINGS (8)
DIZI SEKTÖRÜ KRIZDE YÜZDE... TRACKINGS (5)
TERCIHLERDE IL, ILÇE SINIRI... TRACKINGS (22)
BDP'DEN BAŞBUĞ'A ÇOK SERT... TRACKINGS (11)
İPTAL AYM'NIN VARLIĞINI TARTIŞMAYA... TRACKINGS (10)
KKTC'DE SENDIKALAR YINE KUR'AN... TRACKINGS (20)
AKP'LI BAŞKAN KÜRT SORUNUNA... TRACKINGS (14)
TOKİ'NIN ISTIKRARLI PROJESI ATAKENT... TRACKINGS (6)
ÜÇ BAKANLIKTAN PITBULL GENELGESI... TRACKINGS (37)
PEHLIVANLARIN KAYITLARI BAŞLADI... TRACKINGS (7)
MUCIDIN SIR ÖLÜMÜ: CINAYET... TRACKINGS (5)
NAZARBAYEV'E 'ATATÜRK' TEŞEKKÜRÜ... TRACKINGS (23)
VEKIL İÇLI: HUKUK DIŞILIK... TRACKINGS (8)
'HSYK VE YARGITAY'DA İLHAN... TRACKINGS (25)
BIDEN, İRAN'A İSRAIL GARANTISI... TRACKINGS (11)
'BOMBA YÜKLÜ MINIBÜS'E MÜTALAA... TRACKINGS (6)
BARROSO'NUN GÖZÜ KULAĞI KIBRIS'TA... TRACKINGS (56)

Bilkent University - Information Retrieval Group

News-Chain Discovery - Parameter Selection

This page is to select parameters for news-chain discovery, which is supported by TÜBİTAK project no. 113E249.
Please click on "Start News-Chain Discovery" button after entering parameters.

(Note that news-chain discovery will take several minutes if an old news article is selected.)

Select News Date: 2016-05-13

Select News ID & Title: ID: 1257247, Title: "ABD'nin Libya'da özel kuvvetler bulundurduğ

Enter cosine threshold (between 0.0 and 1.0, default value is given): 0.33

Enter social network (co-occurence) threshold

FIG. 10. Screenshots (*top*: front page, *down*: user interface for parameter selection) from Bilkent News Portal where our framework for story-chain discovery is applied. [Color figure can be viewed at wileyonlinelibrary.com]

menu. The bottom screen is where users enter parameters for the algorithm, such as the input document or similarity threshold values.

We observe that mining a large collection can be time-consuming, as experienced in Radev, Otterbacher, Winkel, and Blair-Goldensohn (2005), and Shahaf et al. (2015). To overcome this scaling problem, we asked the user to enter some keywords about the input document, and hence get a subset of news articles to be processed. A similar approach was also applied in the related studies.

The quality of output chains is input-dependent: selecting low similarity thresholds can result in long and noisy chains. Different input documents may require different parameter values.

Conclusion

We present a framework to discover story chains in a given text collection for an input document. We apply a novel text-mining method that uses a zigzagged search that reinvestigates past documents based on the updated chain. News actors are utilized to reveal connections among news articles. We conducted two user studies that evaluated our framework in terms of effectiveness. The first compares several versions of the framework to set a guideline for use. The second compares our method with three baselines. The results show that our method provides statistically significant improvement in effectiveness, in 61% of pairwise comparisons, with medium or large effect size.

In future work, our framework can be extended and adopted into other domains that use temporal data, such as the analysis of intelligence reports and micro-blogs. Furthermore, there is a need for visualization tools that can help users examine chains, and test collections that can help researchers assess and compare their results.

Acknowledgments

We thank the referees for their constructive criticism, Alper Can for his valuable comments, and our annotators for their support. This work was partly supported by the Scientific and Technical Research Council of Turkey (TÜBİTAK) under grant number 113E249. Any opinions, findings, and conclusions or recommendations expressed in this article belong to the authors and do not necessarily reflect those of the sponsor; therefore, no official endorsement should be inferred.

References

Allan, J. (Ed.). (2002). Topic detection and tracking: Event-based information organization. Norwell, MA: Kluwer Academic.

Allan, J., Papka, R., & Lavrenko, V. (1998). On-line new event detection and tracking. In *Proceedings of the 21st ACM SIGIR conference on research and development in information retrieval (SIGIR'98)* (pp. 37–45). New York: ACM.

Can, F., Kocberber, S., Baglioglu, O., Kardas, S., Ocalan, H.C., & Uyar, E. (2010). New event detection and topic tracking in Turkish. Journal of the American Society for Information Science and Technology, 61, 802–819.

Can, F., Kocberber, S., Balcik, E., Kaynak, C., Ocalan, H.C., & Vursavas, O.M. (2008). Information retrieval on Turkish texts. Journal of the American Society for Information Science and Technology, 59, 407–421.

Choudhary, R., Mehta, S., Bagchi, A., & Balakrishnan, R. (2008). Towards characterization of actor evolution and interactions in news corpora. In *Proceedings of the IR Research, 30th European Conference on Advances in Information Retrieval (ECIR'08)* (pp. 422–429). Berlin: Springer.

Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2nd ed.). New York: Academic Press.

Cohen, J. (1992). A power primer. Psychological Bulletin, 112, 155–159.

Conover, W.J. (1999). Practical nonparametric statistics (3rd ed.). New York: Wiley.

Cucerzan, S. (2007). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)* (pp. 708–716).

Fleiss, J.L. (1971). Measuring nominal scale agreement among many raters. Psychological Bulletin, 76, 378–382.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association, 32, 675–701.

Hossain, M.S., Butler, P., Boedihardjo, A.P., & Ramakrishnan, N. (2012). Storytelling in entity networks to support intelligence analysts. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12)* (pp. 1375–1383). NY: ACM.

Kim, D., & Oh, A. (2011). Topic chains for understanding a news corpus. In *Proceedings of the 12th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing'11)* (pp. 163–176). Berlin: Springer.

Kleinberg, J. (2003). Bursty and hierarchical structure in streams. Data Mining and Knowledge Discovery, 7, 373–397.

Krosnick, J.A., Holbrook, A.L., Berent, M.K., Carson, R.T., Hanemann, W.M., Kopp, R.J., … Moody, W.R. (2002). The impact of "no opinion" response options on data quality: Non-attitude reduction or an invitation to satisfice? Public Opinion Quarterly, 66, 371–403.

Kumaran, G., & Allan, J. (2004). Text classification and named entities for new event detection. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)* (pp. 297–304). NY: ACM.

Küçük, D., & Yazıcı, A. (2011). Exploiting information extraction techniques for automatic semantic video indexing with an application to Turkish news videos. Knowledge-Based Systems, 24, 844–857.

Kumar, D., Ramakrishnan, N., Helm, R.F., & Potts, M. (2008). Algorithms for storytelling. IEEE Transactions on Knowledge and Data Engineering, 20, 736–751.

Landis, J.R., & Koch, G.G. (1977). The measurement of observer agreement for categorical data. Biometrics, 33, 159–174.

Liu, B. (2011). Web data mining: Exploring hyperlinks, contents, and usage data (2nd ed.). Berlin: Springer.

Mei, Q., & Zhai, C. (2005). Discovering evolutionary theme patterns from text: An exploration of temporal text mining. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD'05)* (pp. 198–207). New York: ACM.

Nadeau, D., & Sekine, S. (2007). A survey of named entity recognition and classification. Lingvisticae Investigationes, 30, 3–26.

Nallapati, R., Feng, A., Peng, F., & Allan, J. (2004). Event threading within news topics. In *Proceedings of the 13th ACM International Conference on Information and Knowledge Management (CIKM'04)* (pp. 446–453). NY: ACM.

Radev, D., Otterbacher, J., Winkel, A., & Blair-Goldensohn, S. (2005). NewsInEssence: Summarizing online news topics. Communications of the ACM, 48, 95–98.

Shahaf, D., & Guestrin, C. (2012). Connecting two (or less) dots: Discovering structure in news articles. ACM Transactions on Knowledge Discovery from Data, 5, Article 1-24.

Shahaf, D., Guestrin, C., Horvitz, E., & Leskovec, J. (2015). Information cartography. Communications of the ACM, 58, 62–73.

Shapiro, S.S., & Wilk, M.B. (1965). An analysis of variance test for normality (complete samples). Biometrika, 52, 591–611.

Song, J., Huang, Y., Qi, X., Li, Y., Li, F., Fu, K., & Huang, T. (2016). Discovering hierarchical topic evolution in time-stamped documents. Journal of the Association for Information Science and Technology, 67, 915–927.

Stranieri, A., & Zeleznikow, J. (2011). Knowledge discovery from legal databases (1st ed.). Berlin: Springer.

Subašić, I., & Berendt, B. (2010). Discovery of interactive graphs for understanding and searching time-indexed corpora. Knowledge and Information Systems, 23, 293–319.

Toraman, C., Can, F., & Koçberber, S. (2011). Developing a text categorization template for Turkish news portals. In *Proceedings of 2011 International Symposium on Innovations in Intelligent Systems and Applications (INISTA'11)* (pp. 379–383). IEEE.

Tseng, Y.H., Lin, C.J., & Lin, Y.I. (2007). Text mining techniques for patent analysis. Information Processing & Management, 43, 1216–1247.

Wilcoxon, F. (1945). Individual comparisons by ranking methods. Biometrics Bulletin, 1, 80–83.

Yan, R., Wan, X., Otterbacher, J., Kong, L., Li, X., & Zhang, Y. (2011). Evolutionary timeline summarization: A balanced optimization framework via iterative substitution. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11)* (pp. 745–754). NY: ACM.

Yang, C.C., Shi, X., & Wei, C.P. (2009). Discovering event evolution graphs from news corpora. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 39, 850–863.

Zhu, X., & Oates, T. (2014). Finding story chains in newswire articles using random walks. Information Systems Frontiers, 16, 753–769.

DOI: 10.1002/asi