

Utilizing Temporal Information for Taxonomy Construction

Luu Anh Tuan ^{*1}, Siu Cheung Hui ^{†2}, See Kiong Ng ^{#3}

^{*}*Institute for Infocomm Research, Singapore*

¹`at.luu@i2r.a-star.edu.sg`

[†]*School of Computer Engineering, Nanyang Technological University, Singapore*

²`asschui@ntu.edu.sg`

[#]*Institute of Data Science, National University of Singapore*

³`seekiong@nus.edu.sg`

Abstract

Taxonomies play an important role in many applications by organizing domain knowledge into a hierarchy of ‘*is-a*’ relations between terms. Previous work on automatic construction of taxonomies from text documents either ignored temporal information or used fixed time periods to discretize the time series of documents. In this paper, we propose a time-aware method to automatically construct and effectively maintain a taxonomy from a given series of documents pre-clustered for a domain of interest. The method extracts temporal information from the documents and uses a timestamp contribution function to score the temporal relevance of the evidence from source texts when identifying the taxonomic relations for constructing the taxonomy. Experimental results show that our proposed method outperforms the state-of-the-art methods by increasing F-measure up to 7%-20%. Furthermore, the proposed method can incrementally update the taxonomy by adding fresh relations from new data and removing outdated relations using an information decay function. It thus avoids rebuilding the whole taxonomy from scratch for every update and keeps the taxonomy effectively up-to-date in order to track the latest information trends in the rapidly evolving domain.

1 Introduction

The explosion in the amount of unstructured text data gives us the opportunity to explore knowledge in depth, but there are also challenges to recognize useful information for our interests. To provide access to information effectively, it is important to organize unstructured data in a structured and meaningful manner. Taxonomies, which serve as backbones for structured knowledge, are use-

ful for many NLP applications such as question answering (Harabagiu et al., 2003) and document clustering (Fodeh et al., 2011). However, hand-crafted, well-structured taxonomies such as WordNet (Miller, 1995), OpenCyc (Matuszek et al., 2006) and Freebase (Bollacker et al., 2008), which are publicly available, can be incomplete for new or specialized domains. As it is time-consuming and cumbersome to create a new one manually, methods for automatic domain-specific taxonomy construction from text corpora are highly desirable.

Previous work on automatic construction of domain-specific taxonomies from text documents assumed that the data sets (that is, the document sets) and the underlying taxonomic relations are *static*. However, the data sets for certain domains may evolve over time, as new documents are added while older documents are deleted or modified. As such, the taxonomic relations for these potentially fast-changing domains may not remain static but become *dynamic* over time as new domain terms emerge while some older ones disappear. For example, in World Health Organization reports about *disease outbreak*, the term ‘*smallpox*’ used to be a hyponym of ‘*dangerous disease*’, but it has fallen off since 1980. On the other hand, since 2014 the term ‘*Ebola*’ has become an emerging hyponym of ‘*dangerous disease*’. As another example, up until 1992, in a collection of US yearly reports of terrorism, the term ‘*Palestine Liberation Organization*’ used to be a hyponym of ‘*terrorist group*’, but it is no longer true nowadays. ‘*Palestine Liberation Organization*’ should now be classified as a ‘*national organization*’ of Palestine.

When temporal information in data sets is not captured, the resultant taxonomy may be incomplete or outdated and misleading. This could be caused by the overwhelming evidence of older patterns/contexts compared to emerging, but relatively

small amount of, evidence of newer relations. For example, in the taxonomy of US yearly reports on terrorism, many previous methods could fail to recognize the taxonomic relation between the two terms *'ISIS'* and *'terrorist group'* due to relatively infrequent mentions of *'ISIS'* (which only appears in reports from 2014). Meanwhile, *'Palestine Liberation Organization'* could still be classified as a hyponym of *'terrorist group'* because of the relatively larger number of mentions in the documents from the earlier years.

In this paper, we propose a time-aware method for domain-specific taxonomy construction from a time series of text documents for a particular domain. We incorporate temporal information into the process of identifying taxonomic relations by computing evidence scores for the data sources weighted by a timestamp contribution function (Efron and Golovchinsky, 2011; Li and Croft, 2003) to capture the temporally-varying contributions of evidence from various documents at a particular point in time. We assume that newer evidence is more important than older evidence. For example, the evidence that *'Palestine Liberation Organization'* was a hyponym of *'terrorist group'* in 1990 is less important now than the evidence that *'ISIS'* is a hyponym of *'terrorist group'* in 2014. In the proposed method, we incorporate the timestamp contribution function into the method of Tuan et al. (2014) to measure the weights of the evidence for both statistical and linguistic methods. With such built-in time-awareness for taxonomy construction, we ensure that the constructed taxonomies are up-to-date for the fast-changing domains found in newswire and social media, where users constantly search for updated relations and track information trends.

Most previous work requires re-running the taxonomy construction process whenever there are new incoming data. Our proposed method enables incremental update of the constructed taxonomies to avoid costly reconstructions. We incorporate an information decay function (Smucker and Clarke, 2012) to manage outdated relations in the constructed taxonomy. The decay function measures the extent that the relation is out-of-date over time, and we incorporate it into a time-aware graph-based algorithm for taxonomy update.

The contributions of our research are summarized as follows:

- We propose a time-aware method for taxonomy construction that extracts and utilizes temporal information to measure evidence weights of taxonomic relationships. Our method constructs an up-to-date taxonomy by adding new emerging relations and discarding obsolete and incorrect ones.
- We propose an incremental time-aware graph-based algorithm to update an existing taxonomy, instead of rebuilding a new taxonomy from scratch.

2 Related Work

Previous work on taxonomy construction can be roughly divided into two main approaches: statistical learning and linguistic pattern matching. Statistical learning methods for taxonomy construction include co-occurrence analysis (Lawrie and Croft, 2003), hierarchical Latent Dirichlet Allocation (LDA) (Blei et al., 2004), clustering (Li et al., 2013), term embedding (Tuan et al., 2016), linguistic feature-based semantic distance learning (Yu et al., 2011), and co-occurrence subnetwork mining (Wang et al., 2013). Supervised statistical methods (Petinot et al., 2011) rely on hierarchical labels to learn the corresponding terms for each label. The labeled training data, however, are costly to obtain and may not always be available in practice. Unsupervised statistical methods (Pons-Porrata et al., 2007; Li et al., 2013; Wang et al., 2013) are based on the idea that terms that frequently co-occur may have taxonomic relationships. However, these methods generally achieve low accuracy.

Linguistic approaches for taxonomy construction rely on lexical-syntactic patterns (Hearst, 1992) (e.g., *'A such as B'*) to capture textual expressions of taxonomic relations, matching them with given documents or Web information to identify the relations between a term and its hypernyms (Kozareva and Hovy, 2010; Navigli et al., 2011; Wu et al., 2012). These patterns can be manually created (Wu et al., 2012) or automatically identified (Navigli et al., 2011). Such linguistic pattern matching methods can generally achieve higher precision than the statistical methods, but they suffer from lower coverage. To balance precision and recall, Zhu et al. (2013) and Tuan et al. (2014) have combined both unsupervised statistical and linguistic methods to

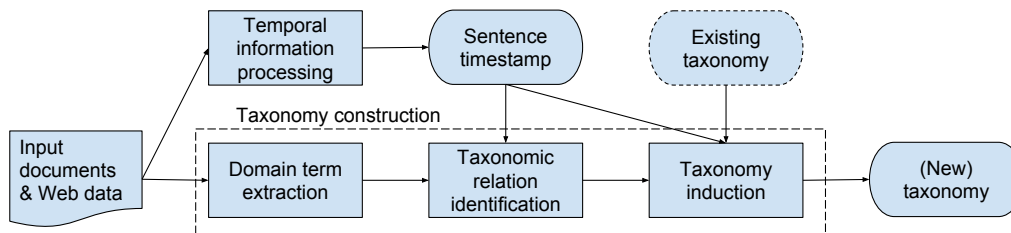


Figure 1: Workflow of the proposed time-aware taxonomy construction method.

find taxonomic relations.

The approach that is closest to our work is the one proposed by Zhu et al. (2013), which performs dynamic taxonomy update. To keep up with ever changing social media data, new terms are mined from incoming data and added to the existing taxonomy. The data are divided into separate clusters using pre-defined time periods based on their document timestamps. The newly found taxonomic relations in each period are then added to the existing taxonomy. The use of a pre-defined time period to discretize the time series of the documents for taxonomy update can be problematic. If the chosen time period is too long, rapid changes of domain terms and their taxonomic relations that have occurred within the time period may not be reported. If the time period is too short, it may fail to identify valid taxonomic relations that needed a longer time period to establish. The method also does not remove those relations in the existing taxonomy that may have become obsolete over time.

3 Problem Specification

We define the root term of a domain-specific taxonomy as a word or phrase that represents the domain of interest. It can be any informative concept such as an entity (*‘animal’*) or event (*‘Ebola outbreak’*). Given a root term \mathcal{R} , we define a corpus \mathcal{C} as a pre-clustered set of a time series of text documents according to \mathcal{R} .

Given two terms t_1 and t_2 , we denote $t_1 \rightarrow t_2$ as a taxonomic relation where t_1 is a hypernym of t_2 . In this work, we define a taxonomy as a triple $\mathcal{H} = (V, E, s)$, where:

- V is the set of the taxonomy’s vertices, i.e., the set of terms, including the root term.
- E is the set of the taxonomy’s edges, i.e., the set of taxonomic relations.
- s is the creation time of the taxonomy. It can be the current date or any specified time.

Our task is formally defined as follows: *Given a root term \mathcal{R} , a corpus \mathcal{C} and an optional existing taxonomy $\mathcal{H}_1 = (V_1, E_1, s_1)$ constructed at time s_1 , we aim to build a new taxonomy $\mathcal{H}_2 = (V_2, E_2, s_2)$ at time s_2 , where $s_2 > s_1$, so that we can process the document set in \mathcal{C} up to time s_2 into the relevant terms in the taxonomy.*

If \mathcal{H}_1 does not exist, the problem becomes creating a taxonomy \mathcal{H}_2 for corpus \mathcal{C} . Otherwise, the problem is to update the existing taxonomy with the newly obtained data for corpus \mathcal{C} . Note that while the taxonomy construction method is not a totally unsupervised method as it does require as input a corpus (\mathcal{C}) pre-clustered by a domain of interest (i.e., \mathcal{R}), the subsequent steps for constructing the taxonomy given the text corpus are unsupervised.

4 Methodology

Figure 1 shows the workflow of the proposed time-aware taxonomy construction method. There are two key processes in the proposed method: temporal information processing and taxonomy construction.

4.1 Temporal information processing

The aim of the temporal information processing process is to generate temporal information (or timestamp for short) for each sentence in the input document or Web data.

Previous taxonomy construction methods (Zhu et al., 2013) only extract temporal information at the document level, i.e., all information in the document has the same timestamp as the document creation date. This assumption, however, is not always correct. Figure 2 shows a sample report about the flight *MH370* created on 30 July 2015. In this report, the timestamp of each sentence is very different from the document creation date. If we were to simply use the temporal information at the document level, the timestamps for the search areas of *MH370* at different periods will be incorrect. Thus, we propose a

- (1) MH370 was an international passenger flight that disappeared on **8 March 2014**, while flying from Kuala Lumpur to Beijing.
- (2) A multinational search effort began in the Gulf of Thailand and the South China Sea, where the flight's signal was lost on surveillance radar, and was extended to the Strait of Malacca.
- (3) Australia took charge of the search effort on **17 March**, when the search shifted to the southern Indian Ocean.
- (4) The current phase of the search is on the northwest of Perth, Australia, which began in **April 2014**.
- (5) Despite the largest search in aviation history, nothing was found until **29 July 2015**, when a marine debris, later confirmed to be a flaperon from Flight 370, was found on Réunion Island.

Figure 2: A sample report about the flight *MH370* created on 30 July 2015.

method to extract timestamps (i.e., temporal expressions) at the sentence level. The method comprises the following three steps:

Document creation date extraction: First, we extract the timestamp at the document level. The text corpus that we are using for this study consists of a collection of reports, scientific publications and Web search results. For the first two types of documents, the timestamp is the document creation date that can be extracted directly from the data source, i.e., the date of the report, or the date of the publication. For Web search results, we use Google advanced search with customized time range which returns the search results together with their creation dates at the beginning of search snippets.

Temporal expression extraction: Next, the second step proceeds to extract all temporal expressions (e.g. “2015 December 31”) in the document. Here, we use SUTime (Chang and Manning, 2012), a library for recognizing and normalizing time expression using a deterministic rule-based method. The output of this step is a list of time expressions, together with their positions in the document.

Sentence timestamp extraction and normalization: Finally, in the third step, we assign each sentence in the document a time expression as follows:

- First, we assign a temporal value τ as the document creation date.
- For each sentence s in the document:
 - If s contains a temporal expression τ_1 , assign τ_1 as the timestamp of s and update $\tau = \tau_1$.
 - Otherwise, assign τ as the timestamp of s .

Note that we use the format ‘YYYY-MM-DD’ for the temporal expression. If the information of DD or MM is missing, it is replaced with the first day

or first month respectively. For example, ‘December 2015’ will be normalized as ‘2015-12-01’. Using the proposed method, sentence (1) and sentence (2) in the example of Figure 2 will have the same timestamp of ‘2014-03-08’, while sentence (3) will have the timestamp of ‘2014-03-17’.

In Section 5.2, we will show that the extraction of timestamps at the sentence level will improve the performance of the proposed taxonomy construction method as compared to the extraction of timestamps at the document level.

4.2 Taxonomy construction

There are three general steps to constructing a taxonomy: domain term extraction, taxonomic relation identification, and taxonomy induction. We make use of the taxonomy construction method of Tuan et al. (2014) for the first step, incorporate timestamps into the second step of identifying taxonomic relations (Section 4.2.2), and propose an incremental taxonomy induction algorithm for the third step (Section 4.2.4). As extraction of domain term extraction does not affect the temporal aspects of taxonomy construction, the first step of domain term extraction is not within the scope of this study. The reader can refer to Tuan et al. (2014) or Zhu et al. (2013) in which linguistic approaches to extract domain terms are discussed. In this paper, we assume that the list of domain terms is available and we will focus only on discussing the second and third steps for taxonomy construction.

4.2.1 Taxonomic relation identification

In this section, we give an overview of the method to identify taxonomic relations proposed in Tuan et al. (2014). Given an ordered pair of two terms t_1 and t_2 , Tuan et al. (2014) calculates the evidence score that $t_1 \rightarrow t_2$ based on the following three methods:

Syntactic contextual subsumption (SCS): This method derives evidence for $t_1 \rightarrow t_2$ from their syntactic contexts, particularly from triples of the form (*subject, verb, object*). It is observed that if the context set of t_1 mostly contains that of t_2 but not vice versa, then t_1 is likely to be a hypernym of t_2 . To implement this idea, the method finds the most common relation (or *verb*) r of t_1 and t_2 , submits the queries “ $t_1 r$ ” and “ $t_2 r$ ” to Web search engine and collects all search results to construct two corpora

$Corpus_{t_1}^\Gamma$ and $Corpus_{t_2}^\Gamma$ for t_1 and t_2 . The syntactic context sets are then created from these contextual corpora using a non-taxonomic relation identification method. The details of $Score_{SCS}(t_1, t_2)$ can be found in Tuan et al. (2014).

Lexical-syntactic pattern (LSP): This method is to find how much more evidence for $t_1 \rightarrow t_2$ is found on the Web than for $t_2 \rightarrow t_1$. Specifically, a list of manually constructed taxonomic patterns (e.g., “ t_2 is a t_1 ”) is queried with a Web search engine to estimate the amount of evidence for $t_1 \rightarrow t_2$ from the Web. The LSP measure is calculated as follows:

$$Score_{LSP}(t_1, t_2) = \frac{\log(|C_{Web}(t_1, t_2)|)}{1 + \log(|C_{Web}(t_2, t_1)|)}$$

where $C_{Web}(t_1, t_2)$ denotes the set of search results.

String inclusion with WordNet (SIWN): This method is to check the evidence for $t_1 \rightarrow t_2$ by using the combination of string inclusion and references in WordNet synsets. $Score_{SIWN}(t_1, t_2)$ is set to 1 if there is such evidence; otherwise, it is set to 0.

Combined evidence: The three scores are then combined linearly as follows:

$$\begin{aligned} Score(t_1, t_2) = & \alpha \times Score_{SCS}(t_1, t_2) \\ & + \beta \times Score_{LSP}(t_1, t_2) \\ & + \gamma \times Score_{SIWN}(t_1, t_2) \end{aligned}$$

If $Score(t_1, t_2)$ is greater than a threshold value, then t_1 is regarded as a hypernym of t_2 .

4.2.2 Incorporating temporal information into taxonomic relation identification

Previous studies of taxonomic relation identification treated all evidence equally, i.e., evidence from 1950 is treated equally with evidence from 2014. This assumption is not always appropriate, as discussed in Section 1. We propose a time-aware method to identify taxonomic relations by incorporating timestamps into the process of finding evidence, using the following timestamp contribution function:

Definition 1 (Timestamp contribution function). *Given a text sentence d with timestamp s_d , the timestamp contribution of d at time s_0 is defined as:*

$$T_d(s_0) = \xi e^{-\xi(s_0 - s_d)}, \quad (1)$$

where ξ is a control rate, $s_0 > s_d$ and $(s_0 - s_d)$ is the time lapse between s_d and s_0 .

Equation (1) describes the timestamp contribution of a sentence at a specific time by using an exponential distribution function T_d . The intuition behind this function is that the evidence of taxonomic relations found in more recent sentences will be of higher relevance than that found in older sentences. This function is inspired by the work of Efron and Golovchinsky (2011), and Li and Croft (2003), in which it was used to effectively rank documents over time intervals.

Using the timestamp contribution function, we incorporate temporal information into the three taxonomic relation identification methods described in Section 4.2.1, as follows:

LSP method: For each search result snippet d in $C_{Web}(t_1, t_2)$ collected from the Web search engine, we calculate the timestamp contribution score of d by using T_d : $T_d(s_0) = \xi e^{-\xi(s_0 - s_d)}$, where s_0 is a chosen specific time (i.e., the time of taxonomy construction) and s_d is the timestamp of d . Note that s_d has to be earlier than s_0 . The unit of time lapse $(s_0 - s_d)$ depends on the nature of corpus and can be, for instance, a day, a month or even a year. For example, if the corpus is from a fast-changing source such as social media, we can set the unit as day to keep up with the change of data on a daily basis. In contrast, for a corpus from slower changing domains such as scientific disciplines, the unit can be a year. The time-aware score for the LSP method is calculated as follows:

$$\begin{aligned} Score_{LSP}^{Time}(t_1, t_2) = & Score_{LSP}(t_1, t_2) \\ & \times \frac{\sum_{d \in C_{Web}(t_1, t_2)} T_d(s_0)}{|C_{Web}(t_1, t_2)|} \quad (2) \end{aligned}$$

In Equation (2), the original LSP evidence score is multiplied by the average timestamp contribution score of all evidence sentences for the taxonomic relation from the Web. If the number of the returned search results is too large, we will use only the first 1,000 results to estimate the average timestamp contribution of evidence.

Note that the total timestamp contribution score of all evidence sentences $\sum_{d \in C_{Web}(t_1, t_2)} T_d(s_0)$ can be considered as the “weighted size” of $C_{Web}(t_1, t_2)$, i.e., we weigh each evidence sentence using Equation (1) and sum all these weights. However, if we use only the “weighted size” of $C_{Web}(t_1, t_2)$ for the time-aware score $Score_{LSP}^{Time}(t_1, t_2)$, there will be some issues. Firstly, the score $Score_{LSP}^{Time}$ will not

be normalized with respect to the number of evidence sentences. This may lead to potential bias due to large amounts of past evidence—if there were an obsolete or incorrect taxonomic relation with many evidence sentences in the past, it may overwhelm the new taxonomic relations which may only have a small number of recent evidence sentences. Secondly, if we normalize the score, the information on the number of evidence sentences, which is important for the *LSP* method to recognize true taxonomic relationships, will be lost. Therefore, we propose to use Equation (2), which combines both information on the number of evidence sentences (embedded inside the original $Score_{LSP}$ score) and the normalized “weighted size” of $C_{Web}(t_1, t_2)$.

SCS method: Similarly, for each search result snippet d in $Corpus_{t_1}^\Gamma$ and $Corpus_{t_2}^\Gamma$, we calculate the timestamp contribution score of d using the function T_d : $T_d(s_0) = \xi e^{-\xi(s_0 - s_d)}$, where s_0 is a specific time and s_d is the timestamp of d . The time-aware score for SCS method is calculated as follows:

$$Score_{SCS}^{Time}(t_1, t_2) = Score_{SCS}(t_1, t_2) \times \left(\frac{\sum_{d_1 \in Corpus_{t_1}^\Gamma} T_{d_1}(s_0)}{|Corpus_{t_1}^\Gamma|} + \frac{\sum_{d_2 \in Corpus_{t_2}^\Gamma} T_{d_2}(s_0)}{|Corpus_{t_2}^\Gamma|} \right) \quad (3)$$

In Equation (3), the original evidence score of $t_1 \rightarrow t_2$ is multiplied by the average timestamp contribution scores of the returned search snippets. Similar to Equation (2), Equation (3) combines both information on the number of evidence sentences (embedded inside the original score $Score_{SCS}$) and the normalized “weighted size” of them.

SIWN method: Because WordNet does not contain information about timestamps, we set:

$$Score_{SIWN}^{Time}(t_1, t_2) = Score_{SIWN}(t_1, t_2) \quad (4)$$

Combined evidence: The final combined evidence score for the time-aware method is calculated as:

$$Score^{Time}(t_1, t_2) = \alpha \times Score_{SCS}^{Time}(t_1, t_2) + \beta \times Score_{LSP}^{Time}(t_1, t_2) + \gamma \times Score_{SIWN}^{Time}(t_1, t_2) \quad (5)$$

If the value $Score^{Time}(t_1, t_2)$ is greater than a threshold value, we extract the relation $t_1 \rightarrow t_2$.

4.2.3 Parameter learning

We need to estimate the optimal values for the parameters α , β and γ which are used in Equation (5). For this purpose, we apply ridge regression (Hastie et al., 2009). First, we use the time-aware method to create taxonomies for the ‘Animal’, ‘Plant’ and ‘Vehicle’ domains using corpora constructed by a bootstrapping method (Kozareva et al., 2008). Then, we ask two annotators to construct gold standard taxonomies of the three domains (see Section 5.2 for more details) and use them to build the training sets. For each pair of terms (t_1, t_2) found in the gold standard taxonomies, its evidence score is estimated as $(\tau+1)$, where τ is the threshold value for $Score^{Time}$. Finally, we use Equation (5) to learn the best combination of α , β and γ using the ridge regression algorithm. Note that we learn the parameters only once and use them subsequently for the other domains.

4.2.4 Incremental taxonomy induction

To avoid reconstructing a taxonomy whenever there is new incoming data, we propose a novel incremental graph-based algorithm to update an existing taxonomy with a given set of taxonomic relations. The proposed algorithm updates a taxonomy automatically over time based on the information decay function defined below.

Definition 2 (Information decay function). *Given a taxonomic relation r , the information decay of r over the period from time s_1 to time s_2 is computed by the information decay function:*

$$D_r(s_1, s_2) = e^{-\lambda(s_2 - s_1)}, \quad (6)$$

where λ is a decay rate and $s_2 > s_1$.

The intuition behind the information decay function is that the evidential value of a relation will decrease over time at an exponential rate.

Given a root node \mathcal{R} , a set of taxonomic relations T and, optionally, an existing taxonomy $\mathcal{H}_1 = (V_1, E_1, s_1)$ created at time s_1 with vertex set V_1 and edge set E_1 , the proposed graph-based algorithm constructs a new taxonomy $\mathcal{H}_2 = (V_2, E_2, s_2)$ created at time s_2 with vertex set V_2 and edge set E_2 . $t_1 \rightarrow t_2$ denotes the edge from t_1 to t_2 in a taxonomy, and $w(t_1 \rightarrow t_2)$ as the weight of this edge (i.e., evidence score). Algorithm 1 consists of four steps:

Step 1: Update existing taxonomy (lines 2 - 4) This step aims to update the existing taxonomy from

Algorithm 1 Taxonomy induction algorithm

Input: \mathcal{R} : root node of taxonomy;
 T : new taxonomic relation set;
 $\mathcal{H}_1 = (V_1, E_1, s_1)$: existing taxonomy created at time s_1 with vertex set V_1 and edge set E_1 ;
Output: $\mathcal{H}_2 = (V_2, E_2, s_2)$: new taxonomy created at time s_2 with vertex set V_2 and edge set E_2 ;

- 1: Set $V_2 = V_1$ and $E_2 = E_1$
- 2: **for** each edge $(t_1 \rightarrow t_2) \in E_2, t_1 \neq \mathcal{R}$ and $t_2 \neq \mathcal{R}$ **do**
- 3: $w(t_1 \rightarrow t_2) = w(t_1 \rightarrow t_2) \times e^{-\lambda(s_2 - s_1)}$
- 4: **end for**
- 5: **for** each relation $(t_1 \rightarrow t_2) \in T$ **do**
- 6: **if** $(t_1 \rightarrow t_2) \in E_2$ **then**
- 7: $w(t_1 \rightarrow t_2) = w(t_1 \rightarrow t_2) + Score^{Time}(t_1, t_2)$
- 8: **else**
- 9: $E_2 = E_2 \cup (t_1 \rightarrow t_2)$
- 10: $w(t_1 \rightarrow t_2) = Score^{Time}(t_1, t_2)$
- 11: **if** $t_1 \notin V_2$ **then**
- 12: $V_2 = V_2 \cup \{t_1\}$
- 13: **end if**
- 14: **if** $t_2 \notin V_2$ **then**
- 15: $V_2 = V_2 \cup \{t_2\}$
- 16: **end if**
- 17: **if** $\nexists (t_3 \rightarrow t_1) \in E_2$ and $t_3 \neq \mathcal{R}$ **then**
- 18: $E_2 = E_2 \cup (\mathcal{R} \rightarrow t_1)$
- 19: $w(\mathcal{R} \rightarrow t_1) = 1$
- 20: **end if**
- 21: **if** $\exists (\mathcal{R} \rightarrow t_2) \in E$ **then**
- 22: $E_2 = E_2 \setminus (\mathcal{R} \rightarrow t_2)$
- 23: **end if**
- 24: **end if**
- 25: **end for**
- 26: edgeFiltering(\mathcal{H}_2);
- 27: graphPruning(\mathcal{H}_2);

time s_1 to s_2 . In this step, the weight of each edge $(t_1 \rightarrow t_2)$ in E_1 (except the edges connected to root \mathcal{R}) is reduced using the information decay function:

$$w(t_1 \rightarrow t_2) = w(t_1 \rightarrow t_2) \times D_{t_1 \rightarrow t_2}(s_1, s_2)$$

Step 2: Add new relations to existing taxonomy

(lines 5 - 25) This step adds new taxonomic relations to the existing taxonomy and updates their weights. It adds each relation $t_1 \rightarrow t_2$ as a directed edge from the parent node t_1 to child node t_2 if this edge does not exist in the existing taxonomy. Otherwise, we update the weight of this edge with a new evidence score. If t_1 does not have any parent node, t_1 will become a child node of root \mathcal{R} . The edge's weight is updated as follows:

$$w(t_1 \rightarrow t_2) = \begin{cases} 1 & \text{if } t_1 = \mathcal{R} \\ w(t_1 \rightarrow t_2) + Score^{Time}(t_1, t_2) & \text{if } t_1 \rightarrow t_2 \in E_1 \\ Score^{Time}(t_1, t_2) & \text{otherwise} \end{cases}$$

The result of this step is a weighted connected graph containing all taxonomic relations with root \mathcal{R} .

Step 3: Edge filtering (line 26) The graph generated in Step 2 contains some edges with low evidence scores. The reason is that some relations in the existing taxonomy can become outdated during the period from s_1 to s_2 (according to the information decay function), and they do not exist in the new relation set. In this step, each edge $t_1 \rightarrow t_2$ in the graph is revisited, and if its weight is lower than the threshold value of $Score^{Time}$, it will be removed from the graph. In the case that t_2 does not have any other parent node except t_1 , t_2 will be deleted from the vertex set, and edges from t_1 to t_2 's children will be added to the edge set with weights that are equal to the weights of the edges from t_2 to t_2 's children. Then, all edges from t_2 to t_2 's children will be removed from the edge set.

Step 4: Graph pruning (line 27) The graph generated in Step 3 is not an optimal tree as it may contain redundant edges or incorrect edges—for example, those edges that form a loop in the graph. This step aims to produce an optimal tree of the taxonomy from the weighted graph in Step 3.

For this purpose, we apply Edmonds' algorithm (Edmonds, 1967) for finding the optimal spanning arborescence for a weighted directed graph. Using this algorithm, we can find a subset of the current edge set that forms a taxonomy where every non-root node has in-degree 1 and the sum of the edge weights is maximized.

5 Performance Evaluation

We have conducted two experiments for performance evaluation. The first experiment evaluates the performance of our proposed time-aware method on constructing a taxonomy from a given list of terms without any prior knowledge (i.e., without any existing taxonomies). The second experiment evaluates the performance of our proposed method on taxonomy update.

5.1 Datasets

We evaluate our method for taxonomy construction based on the following four datasets of document collections obtained from different domains:

- Artificial Intelligence (AI) domain (Navigli et al., 2011): The corpus is about the root term 'Artificial Intelligence', consisting of 4,976 papers extracted from the IJCAI proceedings

from 1969 to 2014 and the ACL archives from year 1979 to 2014.

- MH370 domain: The corpus is about the root term ‘*Issues related to MH370 search*’. MH370 is the missing flight that went down in the ocean on Saturday, 8 March 2014. The corpus is created by querying the Google search engine with the keyword “MH370” from March 08, 2014 to April 30, 2014 and collecting the first 300 documents from the search results each day. After removing duplicates, the corpus contains a total of 12,307 documents.
- Terrorism domain: The corpus is about the root term ‘*Terrorism*’. It contains 293 reports from “Patterns of Global Terrorism (1988-2003)”¹ and “Country Reports on Terrorism (2004-2014)”² of the US state department. Each report contains about 1,500 words.
- Disease domain: The corpus is about the root term ‘*Disease outbreak*’, created by collecting reports from “Disease outbreaks by year from 1996 to 2014”³ of WHO, and the email archive of ProMed⁴ which is an email based reporting system dedicated to reporting on disease outbreaks that affect human health. The corpus contains a total of 25,370 reports/emails.

Parameter settings. For the rapidly changing domain ‘MH370’, we choose ‘day’ as the unit of the time lapse whereas for the other three domains, we use ‘year’ as the time lapse unit. We set the threshold value of $Score^{Time}$ in Equation (5) as 2.2, and the control rate ξ in Equation (1) and decay rate λ in Equation (6) as 0.15. The setting of these parameters will be discussed in Section 5.4.

5.2 Evaluation on taxonomy construction

5.2.1 Experiment

In this experiment, we compare our time-aware taxonomy construction method with other state-of-the-art methods in the task of constructing a new taxonomy from a given list of terms without any prior knowledge (i.e., without any existing taxonomy). Three state-of-the-art methods in the literature are selected for comparison:

¹<http://www.fas.org/irp/threat/terror.htm>

²<http://www.state.gov/j/ct/rls/crt/index.htm>

³<http://www.who.int/csr/don/archive/year/en/>

⁴<http://www.promedmail.org/>

- Zhu’s method (Zhu et al., 2013): It constructs the taxonomy using evidence from multiple sources such as WordNet, Wikipedia and Web search engines. In their method, both statistical and linguistic approaches are used to infer taxonomic relations.
- Kozareva’s method (Kozareva and Hovy, 2010): It constructs the taxonomy using evidence from a Web search engine by matching the search results with a predefined set of syntactic patterns.
- Tuan’s method (Tuan et al., 2014): It is the non time-aware method described in Section 4.2.1. This method ignores temporal information during taxonomy construction.

To evaluate the effectiveness of extracting timestamps at the sentence level (as described in Section 4.1), we also conduct an experiment on a setting that uses the timestamps out the document level (i.e., all evidence in the document will have the same timestamp information as the document creation date). We use the subscript *docstamp* to denote this setting.

5.2.2 Evaluation metric

In this experiment, we evaluate the constructed taxonomies against the manually created gold standard taxonomies. The gold standard taxonomies are created as follows. For each domain, two annotators are employed at the same time to create taxonomies independently using the list of terms obtained from the domain term extraction module, according to the following rules:

- Rule 1 (Relevancy): Every term in the taxonomy should be related to the root term.
- Rule 2 (Appropriateness): Each edge between two terms should be established at the time the taxonomy is created, if their relation is correct and not obsolete. A relation is obsolete if it is invalid at the time of consideration.
- Rule 3 (Hierarchical structure): The gold standard taxonomy of each domain should form a tree, without redundant paths or cycles.

The annotators then compare their constructed taxonomies. A taxonomic relation $t_1 \rightarrow t_2$ is counted as an agreement if and only if both annotators have t_1 and t_2 in their taxonomies, and there is a directed path from t_1 to t_2 . If an annotator has a taxonomic relation with one vertex not

Domain	Number of vertices	Average depth
MH370	257	3.2
AI	925	5.3
Terrorism	312	3.6
Disease	459	4.0

Table 1: Analysis of gold standard taxonomies.

in the other annotator’s taxonomy, it will be considered as a disagreement. After evaluation, the average inter-annotator agreement on edges of the constructed taxonomies between the two annotators is 87% using Cohen’s kappa coefficient measurement. Finally, the two annotators discuss to come up with the gold standard taxonomies. As a result, the number of nodes and average depth of the taxonomies are summarized in Table 1. We use precision, recall and F-measure to measure the performance of taxonomy construction. Let R and R_{gold} be the set of taxonomic relations of our constructed taxonomy and the gold standard taxonomy respectively; then the metrics are given as follows:

$$precision = \frac{|R \cap R_{gold}|}{|R|}; \quad recall = \frac{|R \cap R_{gold}|}{|R_{gold}|};$$

$$F\text{-measure} = 2 \times \frac{precision \times recall}{precision + recall}.$$

5.2.3 Experimental results

The experimental results are given in Table 2 which shows that our time-aware method achieves significantly better performance than Kozareva’s method and Zhu’s method in terms of F-measure (t-test, p-value<0.05). Our method shows slightly lower precision than that of Kozareva’s method due to the SCS method, but much higher recall and F-measure than Kozareva’s method. In contrast, our method shows slightly lower recall but much higher precision and F-measure than Zhu’s method, which is based on statistical methods such as pointwise mutual information and cosine similarity. On average, our time-aware method improves the F-measure by 20% compared to Kozareva’s method, and by 10% compared to Zhu’s method.

Moreover, the incorporation of timestamps into the time-aware method also contributes to better performance as it helps identify new taxonomic relations effectively, while getting rid of obsolete and incorrect relations. As shown from the experimental results, the time-aware method shows significantly better performance than the non time-aware method (i.e. Tuan’s method) in all four domains in terms of

Method	Domain	P	R	F
Kozareva	MH370	89%	36%	51%
Zhu	MH370	60%	59%	59%
Tuan	MH370	85%	49%	62%
Time-aware _{docstamp}	MH370	81%	54%	65%
Time-aware	MH370	86%	57%	69%
Kozareva	AI	90%	37%	52%
Zhu	AI	59%	57%	58%
Tuan	AI	84%	51%	63%
Time-aware _{docstamp}	AI	80%	55%	65%
Time-aware	AI	84%	57%	68%
Kozareva	Terrorism	91%	32%	47%
Zhu	Terrorism	60%	64%	62%
Tuan	Terrorism	86%	49%	62%
Time-aware _{docstamp}	Terrorism	80%	56%	66%
Time-aware	Terrorism	87%	60%	71%
Kozareva	Disease	90%	29%	44%
Zhu	Disease	57%	56%	56%
Tuan	Disease	83%	50%	62%
Time-aware _{docstamp}	Disease	81%	53%	64%
Time-aware	Disease	85%	55%	67%

Table 2: Experimental results for taxonomy construction. P stands for Precision, R for Recall, and F for F-measure.

F-measure (t-test, p-value<0.05). On average, our time-aware method improves the F-measure by 7% compared to Tuan’s method.

We further examine the taxonomic relations identified by the time-aware method but not by the non-time-aware method, and vice versa. We observed that around 91% of relations found by the time-aware method but not by the non-time-aware method are recent relations (i.e., relations found in recent documents), while around 86% of relations found by the non-time-aware method but not by the time-aware method are obsolete relations. The percentage of taxonomic relations that become obsolete in each of the datasets are summarized in Table 3.

Domain	Percentage of obsolete relations
MH370	21%
AI	5%
Terrorism	12%
Disease	7%

Table 3: Percentage of taxonomic relations that become obsolete.

For example, in the Terrorism domain, our method recognizes ‘ISIS’ as a hyponym of ‘terrorist group’, while the three state-of-the-art methods cannot recognize this. In addition, while the other three methods have extracted the outdated taxonomic relation between ‘Palestine Liberation Organization’

and ‘terrorist group’, our method was able to ignore it. The reason is that the three state-of-the-art methods inferred taxonomic relations using co-occurrence frequency, but ‘ISIS’ has only appeared in reports since 2014. The occurrence frequency of ‘ISIS’ was very low compared to ‘Palestine Liberation Organization’ which was mentioned over the past many years. In contrast, by using the timestamp contribution function to better profile the relevance of evidence over time, our method can recognize the recent relationship of ‘terrorist group’ with ‘ISIS’ while getting rid of the obsolete and incorrect relation with ‘Palestine Liberation Organization’.

From the experimental results of the time-aware and time-aware_{docstamp} methods, we also observe that the use of timestamps extracted at the sentence level is more effective than the use of timestamps at the document level. The timestamps extracted at the sentence level can capture more precisely the temporal information of the facts in fast-changing domains than those at the document level. The results showed that the use of sentence-level timestamps can improve the precision and recall of our taxonomy construction method, improving the F-measure by 4% on average, as compared to the use of timestamps at the document level.

5.3 Evaluation on taxonomy update

5.3.1 Experiment

For fast-changing domains, taxonomies should be frequently and quickly updated. In this experiment, we examine how the proposed time-aware method can effectively update the constructed taxonomies over time to keep up with the latest information trends.

We use the case study of the ‘MH370’ domain for this experiment. During the search operation for the missing flight MH370, there were several turning points which can be captured by the following phases (according to well-known news agencies such as CNN, BBC and the New York Times):

- Phase 1 (from March 08, 2014): The flight lost contact with the airport. The search started from the South China Sea and Gulf of Thailand, and was extended to the Strait of Malacca.
- Phase 2 (from March 13, 2014): Images from satellites indicated the plane might have fallen into the Indian Ocean. The search focus was

moved from the South of Sumatra to the South-West of Perth in the Southern Indian Ocean.

- Phase 3 (from March 28, 2014): Estimation of the aircraft’s remaining fuel and the radar track led the search to shift to a new area, the North-West of Perth in the Southern Indian Ocean.

We apply the proposed time-aware method to construct and update the taxonomy for ‘MH370’ incrementally every two days. We compare our time-aware method with the following three methods:

- Zhu’s method (Zhu et al., 2013): It applies a graph-based algorithm to update taxonomies incrementally with timestamp information.
- Baseline 1: The taxonomy is updated with the newly obtained data every two days, but does not use any temporal information in either taxonomic relation identification (Section 4.2.2) or taxonomy induction (Section 4.2.4). Specifically, Step 1 (update existing taxonomy) in Section 4.2.4 is excluded since we are not using any temporal information, so there is no updating of the weights of the existing taxonomy using the decay function.
- Baseline 2: We construct the taxonomies using temporal information every two days, but only with the new documents from these two days. This allows us to evaluate the effect of retiring all the taxonomic relations built from the previous documents instead of the gradual decay approach in our proposed method.

Here we have chosen the time period of two days because the ‘MH370’ domain was a truly fast-changing domain. As we shall see shortly, even using only the new documents within 2 days to build the taxonomy in our baseline method 2, there were new taxonomic relations updated from the latest information (as shown in the example in Figure 4).

5.3.2 Evaluation metric

When constructing the gold standard taxonomies using the same rules described in Section 5.2, we asked the annotators to select for each parent term at most three sub-terms that are most related to it at the time of taxonomy construction. We denote the set of gold-standard taxonomic relations as S_{gold} . In the same way, when applying the methods of taxonomy construction, we select for each parent term

at most three sub-terms with the highest evidence scores. We denote the set of those automatically extracted taxonomic relations as S . We use the following metrics to evaluate the update of taxonomy:

$$precision = \frac{|S \cap S_{gold}|}{|S|}; \quad recall = \frac{|S \cap S_{gold}|}{|S_{gold}|};$$

$$F\text{-measure} = 2 \times \frac{precision \times recall}{precision + recall}.$$

The intuition for limiting the sub-term number to three for the evaluation is that if a taxonomy can keep up with the newly updated data, it should be able to detect the emerging terms and relations and add them to the taxonomy with high evidence scores so that the user can easily observe an emerging trend of information in the domain as it occurs. In addition, the method should also have the capability to remove any obsolete relations in the taxonomy when they are no longer valid.

5.3.3 Experimental results

From the results shown in Figure 3, we can see that our time-aware method achieves the best performance and significantly outperforms the two baseline methods and Zhu’s method in terms of F-measure (t-test, p-value<0.05). One interesting point to observe is that there are two periods when the time-aware method shows much higher F-measure than the baseline methods and Zhu’s method: from March 12 to March 14, and from March 28 to March 30. During these periods, the performance of baseline method 1 (which does not use any timestamp information) and Zhu’s method drops significantly, while our time-aware method’s performance increases slightly.

One plausible explanation is that there are some turning points on March 13 and March 28, which fall within these periods as described above. During these periods, many new terms/relations such as search area, search focus and search device are added to the corpus. Our time-aware method was able to assign higher weights to the new taxonomic relations than the older relations due to their recent timestamps, even though the frequencies of these new relations are fewer than that of the older relations. In contrast, Zhu’s method and baseline method 1 were unable to recognize these new relations due to their relatively low frequencies in the corpus. In addition, incorrect relations in the existing taxonomy were also removed from the new tax-

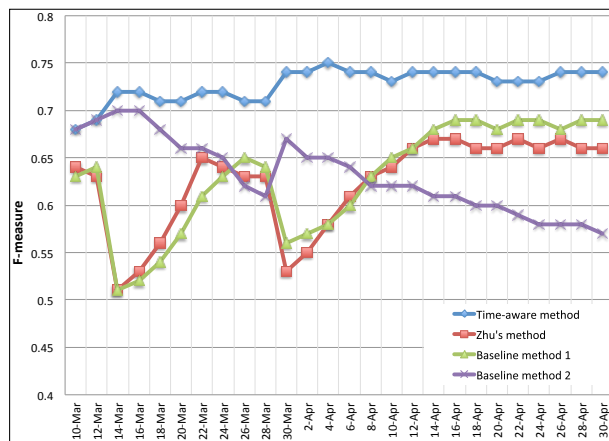


Figure 3: Performance results on taxonomy update over time in the ‘MH370’ domain.

onomy using the information decay function by our time-aware method, whereas the other two methods still kept them in the taxonomy. In short, our time-aware method can update the taxonomy faster with the latest information trends, as well as remove incorrect relations effectively, as compared to the other methods.

Also, from the experimental results of our time-aware method and the baseline method 2, we can observe that updating the existing taxonomy with new taxonomic relations is more effective than rebuilding a new taxonomy using only the new data. The reason is that although some older taxonomic relations are mentioned occasionally in the new data, they are still valid. Therefore, if we ignore the older data, their taxonomic relations will be lost in the new taxonomy when it is constructed with only the new data. In addition, there are also many taxonomic relations that needed a longer time period to become established.

Figure 4 shows an example of the changes of the hyponym list for the term ‘search area’ over time using different methods. We observe that both the time-aware method and baseline method 2, which utilized the temporal information, can quickly update the relations with the latest information as compared to Zhu’s method and baseline method 1, which ignore temporal information for taxonomy construction. For example, in the taxonomy constructed on March 14, the time-aware method and baseline method can quickly recognize the change of the search area to ‘Southern Indian Ocean’ and ‘Sumatra’, thereby ranking them at the top of the hyponym list of ‘search area’, whereas Zhu’s method

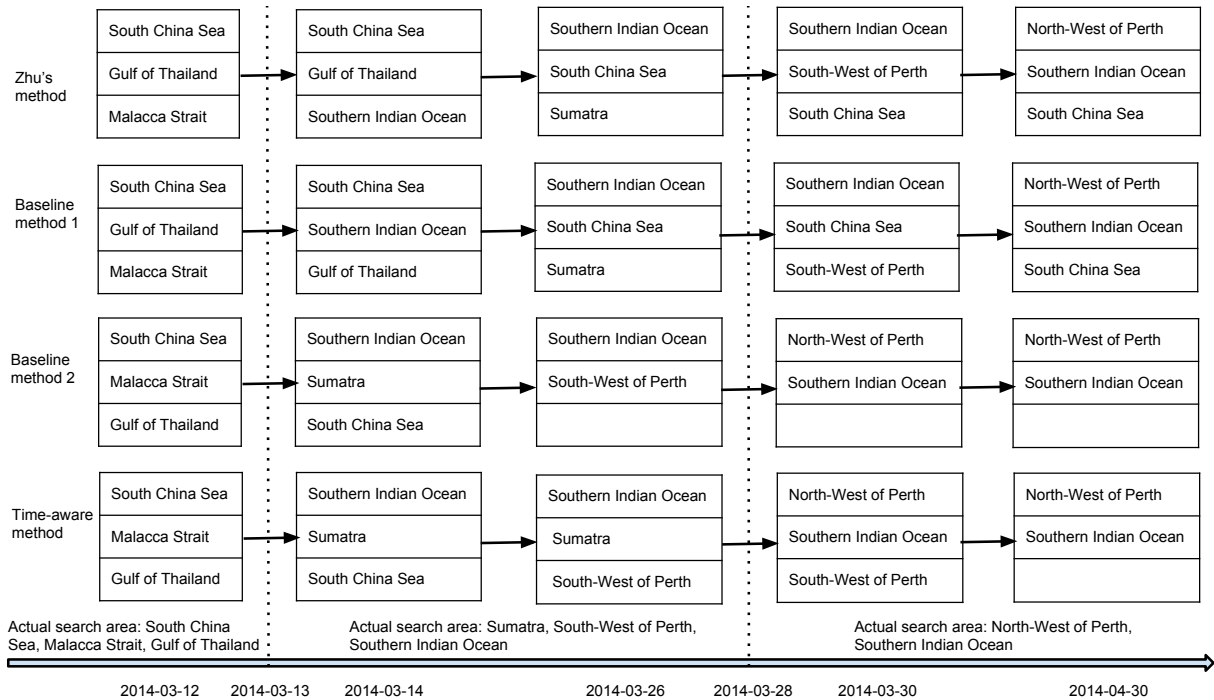


Figure 4: Top three hyponyms of ‘Search area’ in ‘Issues related to MH370 search’ taxonomies over time.

and baseline method 1 both missed this update until March 26. Another interesting point is that due to the lack of temporal information, both Zhu’s method and baseline method 1 still ranked ‘*South China Sea*’ at the top of the taxonomies constructed on April 30, while this term was removed earlier from the hyponym list of ‘*search area*’ by our time-aware method using temporal information.

5.4 Parameter tuning

In our method for taxonomy construction, some parameters are tuned to optimize performance.

The threshold value for $Score^{Time}$ in Equation (5) controls the number of extracted taxonomic relations. In general, the larger this threshold value is, the higher number of true taxonomic relations we can get. However, a higher number of incorrect relations may also occur. From our experiments, we found that the threshold value for $Score^{Time}$ can be set between 2.1 to 2.3 for the time-aware method to achieve the best performance.

The control rate ξ in Equation (1) and decay rate λ in Equation (6) affect the contribution of old and new data. Specifically, smaller values for the control rate and decay rate will allow newer data to contribute more evidence of taxonomic relations than older data, whereas larger values will cause the old and new data to have similar evidence contribu-

tions. According to our experiments, the time-aware method shows the best performance when the values of these rates are set between 0.15 to 0.20.

6 Conclusion

In this paper, we have proposed a novel time-aware method for taxonomy construction given a time series of text documents from a domain that could be fast-changing with emerging concepts or events. By using timestamp contribution and information decay functions, our method can effectively utilize temporal information for both taxonomic relation identification and taxonomy update. The experimental results show that our method achieves better performance than the state-of-the-art methods. In addition, the proposed method can be used to update the taxonomy incrementally over time and keep the taxonomy up-to-date with the latest information trends for the domain. All the datasets, including the gold standards of the four domains and the outputs of our method, are publicly available at <https://sites.google.com/site/tuanluu219/research/tac11>.

Acknowledgements

We are grateful to action editors Lillian Lee and Bo Pang and the anonymous reviewers for their helpful suggestions, which substantially improved the present paper.

References

- David M. Blei, Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. 2004. Hierarchical Topic Models and the Nested Chinese Restaurant Process. *Advances in Neural Information Processing Systems*, pages 17–24.
- Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 1247–1250.
- Angel X. Chang and Christopher D. Manning. 2012. SUTIME: A Library for Recognizing and Normalizing Time Expressions. *Proceedings of the Conference on Language Resources and Evaluation*, pages 3735–3740.
- Jack Edmonds. 1967. Optimum Branchings. *Journal of Research of the National Bureau of Standards B*, 71:233–240.
- Miles Efron and Gene Golovchinsky. 2011. Estimation Methods for Ranking Recent Information. *Proceedings of the 34th ACM SIGIR conference*, pages 495–504.
- Samah Fodeh, Bill Punch, and Pang N. Tan. 2011. On Ontology-driven Document Clustering Using Core Semantic Features. *Knowledge and Information Systems*, 28(2):395–421.
- Sanda M. Harabagiu, Steven J. Maiorano, and Marius A. Pasca. 2003. Open-domain Textual Question Answering Techniques. *Natural Language Engineering*, 9(3):231–267.
- Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. 2009. *The Elements of Statistical Learning*. Springer-Verlag.
- Marti A. Hearst. 1992. Automatic Acquisition of Hyponyms from Large Text Corpora. *Proceedings of the 14th Conference on Computational Linguistics*, pages 539–545.
- Zornitsa Kozareva and Eduard Hovy. 2010. A Semi-supervised Method to Learn and Construct Taxonomies Using the Web. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1110–1118.
- Zornitsa Kozareva, Ellen Riloff, and Eduard H. Hovy. 2008. Semantic Class Learning from the Web with Hyponym Pattern Linkage Graphs. *Proceedings of the 46th Annual Meeting of the ACL*, pages 1048–1056.
- Dawn J. Lawrie and W. Bruce Croft. 2003. Generating Hierarchical Summaries for Web Searches. *Proceedings of the 26th ACM SIGIR conference*, pages 457–463.
- Xiaoyan Li and Bruce W. Croft. 2003. Time-based Language Models. *Proceedings of the 12th ACM CIKM conference*, pages 469–475.
- Baichuan Li, Jing Liu, Chin-Yew Lin, Irwin King, and Michael R. Lyu. 2013. A Hierarchical Entity-based Approach to Structuralize User Generated Content in Social Media: A Case of Yahoo! Answers. *Proceedings of the EMNLP conference*, pages 1521–1532.
- Cynthia Matuszek, John Cabral, Michael J. Witbrock, and John DeOliveira. 2006. An Introduction to the Syntax and Content of CYC. *Proceedings of the AAAI Spring Symposium*, pages 44–49.
- George A. Miller. 1995. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41.
- Roberto Navigli, Paola Velardi, and Stefano Faralli. 2011. A Graph-based Algorithm for Inducing Lexical Taxonomies from Scratch. *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1872–1877.
- Yves Petinot, Kathleen McKeown, and Kapil Thadani. 2011. A Hierarchical Model of Web Summaries. *Proceedings of the 49th Annual Meeting of the ACL*, pages 670–675.
- Aurora Pons-Porrata, Rafael Berlanga-Llavori, and Jose Ruiz-Shulcloper. 2007. Topic Discovery Based on Text Mining Techniques. *Information processing & management*, 43(3):752–768.
- Mark D. Smucker and Charles L. Clarke. 2012. Time-based Calibration of Effectiveness Measures. *Proceedings of the 35th ACM SIGIR conference*, pages 95–104.
- Luu A. Tuan, Jung J. Kim, and See K. Ng. 2014. Taxonomy Construction using Syntactic Contextual Evidence. *Proceedings of the EMNLP conference*, pages 810–819.
- Luu A. Tuan, Yi Tay, Siu C. Hui, and See K. Ng. 2016. Learning Term Embeddings for Taxonomic Relation Identification Using Dynamic Weighting Neural Network. *Proceedings of the EMNLP conference*, pages 403–413.
- Chi Wang, Marina Danilevsky, Nihit Desai, Yinan Zhang, Phuong Nguyen, Thrivikrama Taula, and Jiawei Han. 2013. A Phrase Mining Framework for Recursive Construction of a Topical Hierarchy. *Proceedings of the 19th ACM SIGKDD conference*, pages 437–445.
- Wentao Wu, Hongsong Li, Haixun Wang, and Kenny Q. Zhu. 2012. Probbase: A Probabilistic Taxonomy for Text Understanding. *Proceedings of the ACM SIGMOD conference*, pages 481–492.
- Jianxing Yu, Zheng-Jun Zha, Meng Wang, Kai Wang, and Tat-Seng Chua. 2011. Domain-assisted Product Aspect Hierarchy Generation: Towards Hierarchical Organization of Unstructured Consumer Reviews. *Proceedings of the EMNLP conference*, pages 140–150.
- Xingwei Zhu, Zhao Y. Ming, and Tat-Seng Chua. 2013. Topic Hierarchy Construction for the Organization of

Multi-source User Generated Contents. *Proceedings of the 36th ACM SIGIR conference*, pages 233–242.