Identifying Relevant Temporal Expressions for Real-World Events

Nattiya Kanhabua
L3S Research Center
Leibniz Universität Hannover
Hannover, Germany
kanhabua@L3S.de

Sara Romano
Dipartimento di Informatica e Sistemistica
University Federico II Naples
Naples, Italy
sara.romano@unina.it

Avaré Stewart
L3S Research Center
Leibniz Universität Hannover
Hannover, Germany
stewart@L3S.de

ABSTRACT

Event detection is an interesting task for many applications, for instance: surveillance, scientific discovery, and Topic Detection and Tracking. Numerous works have focused on detecting events from unstructured text and determining what features constitutes an event, e.g., key terms or named entities. Although most works are able to find *interesting* time associated to an event, there is a lack in research on determining the *relevance of time* for an event. In this paper, we propose a method for automatically extracting real-world events from unstructured text documents. In addition, we propose a machine learning approach to identifying relevant time (i.e., temporal expressions) for the extracted events using three classes of features: sentence-based, document-based and corpus-specific features. Through experiments using real-world data and 3,500 manually judged relevance pairs, we show that our proposed approach is able to identify the relevant time of events with good accuracy.

Categories and Subject Descriptors H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval
General Terms Algorithms, Experimentation
Keywords event extraction, public health events, temporal expressions, relevance ranking

1. INTRODUCTION

When did it begin? or *How long will it last?* Such questions related to *time* commonly arise when reading news about a particular event, e.g., wars, political movements, sports competitions, natural disasters or disease outbreaks. The answer to such questions can be regarded as the *temporal fact* about an event, which is defined as a time point for an instantaneous event, or a time span for an event with a known begin and end duration [7]. Temporal facts about an event can be captured by temporal expressions mentioned in documents, i.e., a *time point* and a *time period* as illustrated in the following sentences for two real-world events: the 2011 Arab Spring and the E.coli outbreak in Germany in 2011. (I) *The Arab Spring event was reported to begin in Tunisia on* ***January 11, 2011*** *when demonstrators protested chronic unemployment and police brutality.* (II) *An outbreak of severe illness is causing concern in Germany, where 3 women have died and 276 cases of hemolytic uremic syndrome have been reported since the* ***2nd week of May 2011****.*

Knowing about the temporal facts of an event of interest is useful for both a journalist (in order to write a news story) or a news reader (in order to understand the news). Moreover, temporal facts are also leveraged in many application areas, e.g., answering *temporal questions*, and browsing or querying *temporal knowledge*. Existing work on extracting temporal facts for an event follows two main directions: 1) extract temporal expressions from unstructured text using time and event recognition algorithms [18, 20], and 2) harvest temporal knowledge from semi-structured contents like Wikipedia infoboxes [7]. Unfortunately, previous approaches in the first group have not considered the relevance of temporal expressions, while the latter method is only applicable for the limited number of events with infoboxes provided.

In this paper, we propose an approach to automatically extracting real-world events from unstructured text documents, focusing our study on public health events, i.e., infectious disease outbreaks. To this end, we seek to answer the research question: *how to determine the relevance of temporal expressions for real-world events?* Determining relevant temporal expressions is a challenging task since there can be many temporal expressions associated to an event and not all of them are equally relevant. Hence, we propose a *machine learning approach* to identifying relevant temporal expressions for a given event using three classes of features: document-based, sentence-based, and corpus-specific.

The main contributions of this paper are: 1) an approach to automatically extracting real-world events from unstructured documents, 2) a machine learning approach to identifying the relevant temporal expressions of the extracted events, 3) proposing three classes of features for learning the relevance of time, and 4) extensive experiments on real-world data and 3,500 manually judged relevance pairs.

The organization of the rest of the paper is as follows. In Section 2, we give an overview of related work. In Section 3, we outline our system architecture. In Section 4, we explain our document- and event model. In Section 5, we propose a method for extracting events from unstructured documents. In Section 6, we propose an approach to identifying relevant time for a given event. In Section 7, we evaluate our proposed approach. Finally, we conclude our work in Section 8.

2. RELATED WORK

A number of ranking models exploiting temporal information have been proposed, including [1, 4, 11, 12]. Li and Croft [11] incorporated time into language models, called time-based language models, by assigning a document prior using an exponential decay function of a document creation

date. They focused on recency queries, such that the more recent documents obtain the higher probabilities of relevance. Diaz and Jones [4] used document creation dates to measure the distribution of retrieved documents and create the temporal profile of a query. They showed that the temporal profile together with the contents of retrieved documents can improve average precision for the query by using a set of different features for discriminating between temporal profiles. Berberich et al. [1] integrated temporal expressions into query-likelihood language modeling, which considers time uncertainty inherent to a query and documents, i.e., temporal expressions can refer to the same time interval even they are not exactly equal. Metzler et al. [12] considered implicit temporal information needs. They proposed mining query logs and analyze query frequencies over time in order to identify time-sensitive queries.

There are also works that have focused on recency ranking [2, 5, 6, 8], while analyzing queries over time has been studied in [10, 16]. Kulkarni et al. [10] studied how users' information needs change over time, and Shokouhi [16] employed different time series analysis methods for detecting seasonal queries. For an entity-ranking task, Demartini et al. [3] analyzed news history (i.e., past related articles) for identifying relevant entities in current news articles. Kanhabua et al. [9] introduced the task of *ranking related news predictions* with the main goal of improving information access to predictions most relevant to a given news story.

The most relevant work to us is presented by Strötgen et al. [17], where they studied the problem of identifying *top relevant temporal expressions* in documents. Note that, our work differs from the previous work in some aspects. We focus on identifying relevant temporal expressions with respect to a particular event (namely, a public health event or infectious disease outbreak), which is assumed as a mention of both *named entity* and *place*. On the other hand, the previous work considered the relevance of temporal expressions in general, or with respect to a given query.

3. SYSTEM OVERVIEW

As depicted in Figure 1, our system consists of two major phases: 1) text annotation, and 2) event extraction.

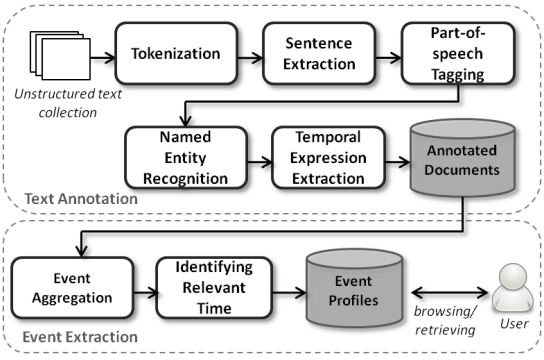

Figure 1: System architecture.

We automatically extract real-world events in a pipeline fashion. The text annotation phase consists of: tokenization, sentence extraction, part-of-speech (dependency) tagging, named entity recognition and temporal expression extraction. The result of this step is a set of documents annotated with named entities and temporal expressions, which will be used for extracting events.

The next step is to extract events from annotated documents in the previous step. We assume that an event is described as a sentence containing mentions of a disease name and a location (further details are given in Section 5). The time of an event is determined as a temporal expression mentioned in the *same sentence* or its *surrounding context sentences*, where a temporal expression is considered **relevant** to a given event if it refers to the *starting-*, *ending-*, or *ongoing* time of the event. The results from this step are a set of events associated to a given named entity and occur in particular time and place (denoted as the *event profiles* of a named entity).

4. MODEL

A document collection used for extracting events can be news articles or official reports about events of interest. The collection is composed of unstructured text documents: $C = \{d_1, \ldots, d_n\}$. A document d is represented as a bag-of-words or an unordered list of terms: $d = \{w_1, \ldots, w_k\}$. Each document is associated to an *annotated document* $\hat{d}$ consisting of three components: $\hat{d}_{ne}$, $\hat{d}_t$, $\hat{d}_s$. The component $\hat{d}_{ne}$ represents a set of named entities: $\hat{d}_{ne} = \{ne_1, \ldots, ne_k\}$. In this work, we are interested in named entities relevant to the medical domain, i.e., *diseases*, *victims*, and *locations*. A disease is represented by a named entity of the type *medical condition*, while a victim is recognized as a named entity in the categories *population*, *age*, *family*, *animal*, *food*, and *plant*. Finally, a location is represented by a geographic expression. The component $\hat{d}_t$ is a set of temporal expressions mentioned in d: $\hat{d}_t = \{t_1, \ldots, t_h\}$. The component $\hat{d}_s$ is a partially ordered set of sentences contained in d: $\hat{d}_s = \left\{s_1, \ldots, s_z \mid \bigcup_{j=1}^{z} s_j = d\right\}$ where $\forall i, j = 1, \ldots, z \; s_i \leq_s s_j$ indicating the sentence s_i precedes s_j in d.

An event corresponds to a real-world outbreak event defined as a quadruple: $e = (v, m, l, t_e)$ described by four attributes that provide information on *who* (victim v) was infected by *what* (disease or medical condition m), *where* (location l) and *when* (time t_e). These four features of an event are extracted from a set of annotated documents.

We distinguish two types of temporal information associated to an event e: 1) the publication time of a document d reporting about e, and 2) content time indicating when an event e has actually taken place. For instance, it was reported by the World Health Organization (WHO) on **29 July 2012** about an ongoing Ebola outbreak in Uganda since the **beginning of July 2012**. In this case, the time of the event is the **beginning of July 2012**, whereas the time when the event was reported is **29 July 2012**.

5. EVENT EXTRACTION

Our approach to event extraction in the domain of infectious disease outbreaks is based on a simplified assumption, i.e., an event can be described as the co-occurring of two entity types (*medical condition* and *geographic expression*). This simplification is based on a minimum requirement by domain experts to assess a public health event.

A set of events will be extracted from each annotated document $\hat{d} \in C$, where an event candidate is a sentence $s_i \in \hat{d}_s$ containing **both** a medical condition entity m and a geographic expression l. In other words, we consider a pair of m and l to form the basis for an event if they occur in the same sentence s_i. This assumption is more precise than forming events from the cross product of all pairs of medical conditions and locations in a documents which can result in high

false positives [19]. Note that, there can be more than one geographic expression mentioned in s_i. In this work, we are interested in *country-level* geographic information. Thus, geographic expressions with finer granularity levels (e.g., cities, provinces and states) will be normalized by mapping them to the *country-level* granularity using a geo-tagger tool. Finally, we will assume that the identified event is associated to all geographic expressions mentioned in s_i. The results of this step is a set of *event candidates*: $E_C = \{s_1, \ldots, s_k\}$, where each $s_i \in E_C$ is a sentence associated to a pair of medical condition m and location l.

The next step is to identify *victims* and *time* associated to each event candidate $s_i \in E_C$. We determine victims as named entities and (*possibly relevant*) time as temporal expressions in s_i itself, or in its surrounding context sentences of s_i, i.e., the sentences s_{i-1} and s_{i+1}. The idea of using context sentences is to increase recall of events being discovered. Our simple assumption about event time might not guarantee that all temporal expressions determined are *relevant* to an event. If **no** victim and temporal expression can be identified for an event candidate s_i, then we will discard s_i from the set of event candidates. To this end, the final results from the event extraction step are a set of event candidates $\{e_1, \ldots, e_q\}$, where each event e_i is represented by a quadruple: $e = (v, m, l, t_e)$ described by four attributes that provide information on *who* v (victim) was infected by *what* (disease or medical condition m), *where* (location l) and *when* (time t_e).

6. IDENTIFYING RELEVANT TIME

The task of identifying relevant time can be regarded as a *classification problem*. That is, we will determine whether a temporal expression is **relevant** or **irrelevant** for a given event. We employ a machine learning method for learning the relevance of temporal expressions using three classes of features: sentence-based, document-based and corpus-specific features. Note that, our proposed features can be applied for the similar task in a generic domain as well.

6.1 Sentence-based Features

Given a temporal expression t_e, the values of features are determined from the sentence s_i containing t_e, where s_i is extracted from an annotated document $\hat{d}$. For this class, we propose 13 features, namely, *senLen*, *senPos*, *isContext*, *cntEntityInS*, *cntTExpInS*, *cntTPointInS*, *cntTPeriodInS*, *entityPos*, *entityPosDist*, *TExpPos*, *TExpPosDist*, *timeDist*, and *entityTExpPosDist*. The intuition is to determine the relevance of temporal expressions by considering *the degree of relevance of their corresponding sentences* with respect to a given event. For example, a sentence that is too long or too short is likely to be less relevant, and a sentence containing too many of geographic expressions is possibly irrelevant or less specific to an event. The granularity type of time mentioned in a sentence can also indicate the relevance to a particular event, e.g., a time point should be more relevant than a time period because it is more precise/accurate.

The first feature *senLen* is a score of the length (in characters) of s_i normalized by the maximum sentence length in $\hat{d}$. The feature *senPos* gives a score of the position of s_i in $\hat{d}$ normalized by the total number of sentences in $\hat{d}$. The feature *isContext* indicates whether s_i is a context sentence or not. The feature *cntEntityInS* is a score of the number of occurrences of entities in s_i normalized by the maximum number of entities in any sentence $s_i \in \hat{d}$. The feature *cntTExpInS* is a score of the number of temporal expressions in s_i normalized by the maximum number of temporal expressions in any sentence $s_i \in \hat{d}$. The features *cntTPointInS* and *cntTPeriodInS* are scores of the numbers of time points and time periods in s_i normalized by the number of temporal expressions in s_i respectively. In other words, the two features take into account time granularities, such as, either a point- or a period of time.

The feature *entityPos* is an average of scores of the positions (in character) of entities in s_i normalized by the length of s_i. The feature *entityPosDist* is an average of scores of the position distance between all pairs of entities in s_i normalized by the length of s_i. The feature *TExpPos* is an average of scores of the positions (in character) of temporal expressions in s_i normalized by the length of s_i. The feature *TExpPosDist* is an average of scores of the position distance between all pairs of temporal expressions in s_i normalized by the length of s_i. The feature *timeDist* is an average of scores of the distance *in time* for all pairs of temporal expression in s_i. Our assumption is that the further distance two time expressions have, the less they are related. The final feature is *entityTExpPosDist*, which is an average of scores of the position distance between all pairs of (entity, time) in s_i normalized by the length of s_i. Note that, this feature is only applicable when s_i is a *not* context, but the original sentence of an event. The value of all features is normalized to range from 0 to 1.

6.2 Document-based Features

We propose five features that are determined at the document level, namely: *cntEntityInD*, *cntEntitySen*, *cntTExpInD*, *cntTPointInD*, *cntTPeriodInD* In general, the proposed features are aimed at capturing the **ambiguity** of a document mentioning about a given event. These features can be computed off-line because they are independent from an event of interest.

The first feature *cntEntityInD* is a score of the number of occurrences of entities in $\hat{d}$ normalized by the total number of sentences in $\hat{d}$. The feature *cntEntitySen* is a score of the number of sentences containing at least one entity normalized by the total number of sentences in $\hat{d}$. The feature *cntTExpInD* is a score of the number of temporal expressions in $\hat{d}$ normalized by the total number of sentences in $\hat{d}$. The feature *cntTPointInD* is a score of the number of time points in $\hat{d}$ normalized by the total number of temporal expressions in $\hat{d}$. The feature *cntTPeriodInD* is a score of the number of time periods in $\hat{d}$ normalized by the total number of temporal expressions in $\hat{d}$. Similar to the previous class, the values of all features is normalized to range from 0 to 1.

6.3 Corpus-specific Features

This class of features is based on heuristics with respect to a particular document collection. Temporal expressions are considered **non-relevant** if they are mentioned in question or negative sentences as well as those related to commercial or vaccinate campaigns. We manually build *negative keywords* corresponding to different aspects mentioned above. The feature *isNeg* is 1 if a sentence s_i contains any term in negative keywords, and it is 0 otherwise. In addition, temporal expressions are **not** related to a given event if they refer to a history of an outbreak, e.g., some statistics in the past. Thus, the feature *isHistory* is 1 if a sentence s_i contains any term related to historical data (e.g., "statistic", "annually", "past year") and it is 0 otherwise.

7. EXPERIMENTS

Our document collection consists of official medical reports posted all over the year 2011 and provided by two different authorities: the World Health Organization [21] and ProMED-mail [15]. The reports contain information about outbreaks and public health treats, which were moderated by medical professionals worldwide. The number of documents and sentences collected for ProMED-mail are 2,977 documents and 95,465 sentences; whereas for WHO only 59 documents were reported resulting in 761 sentences. The text annotation required a series of language processing tools, including OpenNLP [14] (for tokenization, sentence splitting and part-of-speech tagging), OpenCalais [13] (for named entity recognition) and HeidelTime [18] (for temporal expression extraction).

Our dataset was created by manually selecting 25 infectious diseases (medical conditions) by medical professionals, and outbreak events were extracted with respect to the selected diseases. Our main goal is to evaluate the proposed method for identifying the relevant temporal expressions of a given event. Thus, we asked human assessors to evaluate event/time pairs (e.g., relevant or non-relevant) using 3 levels of relevance: 1 for *relevant* to an event, 0 for *irrelevant* to an event or *incorrect* tagged time, and -1 for *unknown*. The *incorrect* tagged time is an error produced by the annotation tools. More precisely, an assessor was asked to give a relevance score $Grade(e, t_e)$ where $Grade(e, t_e)$ is a pair of an event e, and a temporal expression t_e. When t_e is a time period, i.e., containing two dates, an assessor has to give judges to both dates. Hence, an event/time pair (e, t_e) is relevant if and only if there is at least one relevant date, and it is **non-relevant** if all dates are non-relevant. Finally, assessors evaluated about 3500 event/time pairs[1].

The Weka implementation [22] was used for modeling the relevant time identification as a classification task, which was learned using several algorithms: decision tree (J48), Naïve Bayes (NB), neural network (NN) and SVM, using 10-fold cross-validation with 10 repetitions. We measured statistical significance using a t-test with $p < 0.05$. In the table, bold face indicates statistically significant difference from the respective baseline.

Classification results. The baseline method for relevant time classification is the majority classifier. The accuracy of the baseline is 0.58. Table 1 shows the accuracy of different classification algorithms on each feature. The combination of all features within a particular class is denoted *senBased*, *docBased*, and *corpusBased* respectively. *ALL* is the combination of all features among different classes.

The overall results show that decision tree (J48) is the best among other classification algorithms. In general, many of sentence-based features improved the accuracy of baseline significantly. The features *senLen* and *entityPosDist* perform best with accuracy=0.65. While the features in document-based class obtained high accuracy, but they did not significantly improve the baseline. The worst performing features are those from corpus-specific class. The combination of different features gained high accuracy but did not significantly outperform the baseline.

8. CONCLUSIONS

In this paper, we propose a method for automatically extracting real-world events from unstructured text documents.

[1]Available at `http://www.idi.ntnu.no/~nattiya/data/RelTime.zip`.

Table 1: Accuracy of relevant time identification.

Feature	*J48*	*NB*	*NN*	*SVM*
senLen	**.65**	.59	.58	.58
senPos	**.62**	.58	.58	.58
isContext	.58	.58	.58	.58
cntEntityInS	.59	.53	.58	.57
cntTExpInS	.61	.55	.58	.57
cntTPointInS	.60	.59	.59	.58
cntTPeriodInS	.60	.59	.59	.58
entityPos	.65	.58	.58	.57
entityPosDist	**.65**	.58	.58	.57
TExpPos	.58	.58	.58	.58
TExpPosDist	**.59**	.59	.59	.58
timeDist	.58	.58	.49	.58
entityTExpPosDist	.57	.58	.58	.58
cntEntityInD	.62	.58	.58	.57
cntEntitySen	.62	.58	.58	.57
cntTExpInD	.63	.52	.58	.57
cntTPointInD	.61	.58	.58	.58
cntTPeriodInD	.61	.58	.58	.58
isNeg	.58	.58	.58	.58
isHistory	.58	.58	.58	.58
senBased	.66	.55	.59	.59
docBased	.68	.58	.61	.60
corpusBased	.58	.58	.58	.58
ALL	.69	.55	.61	.63

In addition, we propose a machine learning approach to determining the relevance of temporal expressions associated to a given event. Our proposed features are based on annotated documents and domain-specific heuristics. Through experiments using real-world dataset, we show that our proposed approach is able to identify relevant temporal expressions for an event with good accuracy.

Acknowledgments This work is partially supported by the EU Project M-Eco Medical Ecosystem (grant No.247829).

9. REFERENCES

[1] K. Berberich, S. Bedathur, O. Alonso, and G. Weikum. A language modeling approach for temporal information needs. In *Proceedings of ECIR'2010*, 2010.

[2] N. Dai, M. Shokouhi, and B. D. Davison. Learning to rank for freshness and relevance. In *Proceeding of SIGIR'2011*, 2011.

[3] G. Demartini, M. M. S. Missen, R. Blanco, and H. Zaragoza. Taer: time-aware entity retrieval-exploiting the past to find relevant entities in news articles. In *Proceedings of CIKM'2010*, 2010.

[4] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In *Proceedings of SIGIR'2004*, 2004.

[5] A. Dong, R. Zhang, P. Kolari, J. Bai, F. Diaz, Y. Chang, Z. Zheng, and H. Zha. Time is of the essence: improving recency ranking using twitter data. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 331–340, New York, NY, USA, 2010. ACM.

[6] J. L. Elsas and S. T. Dumais. Leveraging temporal dynamics of document content in relevance ranking. In *Proceedings of WSDM'2010*, 2010.

[7] J. Hoffart, F. Suchanek, K. Berberich, and G. Weikum. Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial Intelligence Journal, Special Issue on Wikipedia and Semi-Structured Resources*, 2012.

[8] A. Jatowt, Y. Kawai, and K. Tanaka. Temporal ranking of search engine results. In *Proceedings of WISE*, 2005.

[9] N. Kanhabua, R. Blanco, and M. Matthews. Ranking related news predictions. In *Proceeding of SIGIR'2011*, 2011.

[10] A. Kulkarni, J. Teevan, K. M. Svore, and S. T. Dumais. Understanding temporal query dynamics. In *Proceedings of WSDM'2011*, 2011.

[11] X. Li and W. B. Croft. Time-based language models. In *Proceedings of CIKM'2003*, 2003.

[12] D. Metzler, R. Jones, F. Peng, and R. Zhang. Improving search relevance for implicitly temporal queries. In *Proceedings of SIGIR'2009*, 2009.

[13] OpenCalais, `http://www.opencalais.com/`.

[14] OpenNLP, `http://opennlp.apache.org/`.

[15] ProMED-mail, `http://www.promedmail.org/`.

[16] M. Shokouhi. Detecting seasonal queries by time-series analysis. In *Proceeding of SIGIR'2011*, 2011.

[17] J. Strötgen, O. Alonso, and M. Gertz. Identification of top relevant temporal expressions in documents. In *Proceeding of the 2nd Temporal Web Analytics Workshop (TempWeb02)*, 2012.

[18] J. Strötgen and M. Gertz. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, 2010.

[19] J. Strötgen, M. Gertz, and C. Junghans. An event-centric model for multilingual document similarity. In *Proceeding of SIGIR'2011*, 2011.

[20] M. Verhagen, I. Mani, R. Sauri, J. Littman, R. Knippen, S. B. Jang, A. Rumshisky, J. Phillips, and J. Pustejovsky. Automating temporal annotation with TARSQI. In *Proceedings of ACL'2005*, 2005.

[21] WHO disease outbreak reports, `http://www.who.int/csr/don/en/`.

[22] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition.* Morgan Kaufmann, 2005.