Extracting Personal Medical Events for User Timeline Construction using Minimal Supervision

Aakanksha Naik
Language Technologies Institute
Carnegie Mellon University
anaik@cs.cmu.edu

Chris Bogart
Institute for Software Research
Carnegie Mellon University
cbogart@cs.cmu.edu

Carolyn Rose
Language Technologies Institute
Carnegie Mellon University
cprose@cs.cmu.edu

Abstract

In this paper, we describe a system for automatic construction of user disease progression timelines from their posts in online support groups using minimal supervision. In recent years, several online support groups have been established which has led to a huge increase in the amount of patient-authored text available. Creating systems which can automatically extract important medical events and create disease progression timelines for users from such text can help in patient health monitoring as well as studying links between medical events and users' participation in support groups. Prior work in this domain has used manually constructed keyword sets to detect medical events. In this work, our aim is to perform medical event detection using minimal supervision in order to develop a more general timeline construction system. Our system achieves an accuracy of 55.17%, which is 92% of the performance achieved by a supervised baseline system.

1 Introduction

In recent years, the steady shift towards a consumer-centric paradigm in healthcare, in conjunction with the meteoric rise of social networking, has led to the establishment of several online support groups and an increasing amount of available patient-authored text. Analyzing this text can provide us an opportunity to study many important issues such as how important medical events affect people's lives and how important changes in their personal lives affect disease progression. We can also study how important medical events affect users' participation in these online communities.

To perform such analyses on large-scale data, there is a need to develop automated methods to extract important personal medical events and associate them with dates from user posts in online support groups. These extracted events and dates can then be used to construct medical event timelines for users and study links between user participation or posting behaviors in online support groups and important personal medical events (Wen and Rosé (2012)). Such automated methods can also be used for patient health monitoring. In this work, we propose a novel unsupervised approach to personal medical event extraction that achieves an accuracy of 55.17%, which is 92% of the performance of the most similar supervised approach on a cancer support forum corpus.

Prior work in personal medical event extraction (Wen and Rosé (2012)) from user posts uses manually constructed sets of keywords to detect medical events from text. This limits the generality of such systems, since using the system on a new corpus requires prior knowledge about types of medical events, and the vocabulary used by users to describe these events. To make them more general, we propose a data-driven personal medical event extraction pipeline which detects medical events with minimal supervision. This makes our system independent of the corpus on which it is used and reduces the manual effort required. We test the performance of our system on the task of constructing cancer event timelines from the dataset used by Wen and Rosé (2012). In spite of being almost completely unsupervised, our system reaches 92% of the performance achieved by a supervised baseline system.

The rest of paper is organized as follows. Section 2 describes prior work in event extraction and temporal resolution which we leverage, while section 3 describes our datasets. Section 4 introduces the architecture of our proposed system and section 5 talks about the system modules in more detail. Section 6 describes our experiments and evaluation, while section 7 presents a brief error analysis and describes possible future extensions. Section 8 concludes the paper.

2 Related Work

Event extraction is a well-studied topic in natural language processing. This has resulted in the development of several off-the-shelf tools for event extraction (Saurí et al. (2005), Chambers (2013), Derczynski et al. (2016)). All these tools have been developed for extraction of public events from news corpora. Some prior work has also studied extraction of public events from social media (Sakaki et al. (2010), Becker et al. (2010), Ritter et al. (2012)). However, in this work, we want to focus on extracting personal medical events for users from their posts on online support groups using minimal supervision.

There has been some prior work on personal event extraction from social media, especially twitter)Li and Cardie (2014); Li et al. (2014)). Li et al. (2014) developed a system for personal event extraction from twitter using minimal supervision. They used the presence of congratulations/ condolence speech acts to detect personal event mentions in tweets and clustering based on the Latent Dirichlet Algorithm (Blei et al. (2003)) to detect personal event types. However, they did not focus specifically on medical events. While we also want to build a system for personal event extraction from online support groups, our focus is on identifying medical events. Hence, the techniques used by Li et al. (2014) do not work very well for us. Online support groups are not as person-focused as twitter, so the presence of congratulations/ condolence speech acts is not a strong signal for personal medical event detection. Moreover, as we show in section 6, LDA is unable to perform well on personal medical event type detection. So, we use a different technique for event type detection, which is partly similar to the technique used by Huang et al. (2016). Our overall system pipeline for data-driven medical event detection with minimal supervision is partly inspired by Li et al. (2014).

On the other hand, there has not been extensive research on personal medical event extraction from online support groups. Wen and Rosé (2012) developed a system for medical event extraction from online support groups. Their system used manually constructed keyword sets for event extraction. We propose a minimally supervised medical event detection pipeline which can remove the need to create these manual keyword sets.

Since we want to create event timelines for users from their posts in online support groups, we also need to perform temporal expression detection and resolution as well as linking of temporal expressions to events. Temporal expression extraction and normalization is also a well-studied area and several off-the-shelf systems are available (Strötgen and Gertz (2010), Chang and Manning (2012), Derczynski et al. (2016)). Moreover, some systems perform both temporal resolution and linking of events with temporal expressions (Chambers (2013)). However, most of these systems are developed for news data and do not work very well with the informal writing style used on social media. But there have been some efforts to develop systems which work well for this space. Wen et al. (2013) developed a temporal tagger and resolver for informal temporal references on social media, but the system is not available for use. The HeidelTime system Strötgen and Gertz (2010) also has a "colloquial english" setting which works well for temporal resolution from social media data. To link events with temporal expressions, we use the heuristics proposed by Wen and Rosé (2012).

3 Dataset

We use two datasets in this paper. The first dataset comprises of all posts from two groups called "Knitters with Breast Cancer" and "Beginners Knit-Along" from Ravelry[1], a website for fiber arts enthusiasts. "Knitters with Breast Cancer" is one of the largest and most active breast cancer groups on Ravelry. This group was started in December 2008. As of December 2016, it had 426 members and 120,000 posts. "Beginners Knit-Along" is a group for knitting enthusiasts who have just started learning how to knit. This group was started in 2013. As of December 2016, it had 3274 members and 70,000 posts. The data from these groups is used to create a list of medical terms, based on vocabulary difference, which is used in the medical event extraction module. We

[1] https://www.ravelry.com/

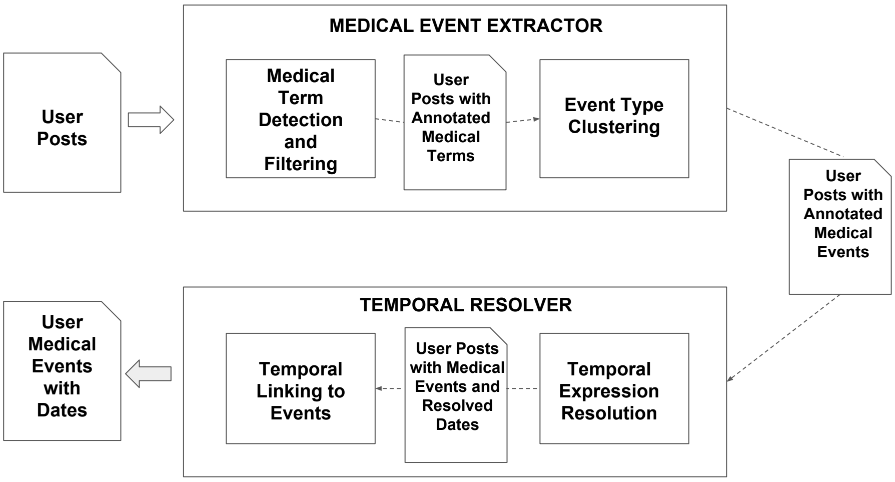
MEDICAL EVENT EXTRACTOR
User Posts
Medical Term Detection and Filtering
User Posts with Annotated Medical Terms
Event Type Clustering
User Posts with Annotated Medical Events
TEMPORAL RESOLVER
User Medical Events with Dates
Temporal Linking to Events
User Posts with Medical Events and Resolved Dates
Temporal Expression Resolution

Figure 1: Architecture of proposed system pipeline

do not use this data for system evaluation.

In order to facilitate comparison with previous work, we use a different dataset to evaluate the performance of our system on the user timeline construction task. This dataset comprises of user posts from an online breast cancer community called breastcancer.org. This dataset is a subset [2] of the annotated dataset used by Wen and Rosé (2012). It comprises of major cancer events and associated dates for 50 users, along with all posts by these users. This dataset is much smaller in comparison to Knitters with Breast Cancer, comprising only of 3293 posts.

4 System Description

Our system pipeline is similar to the pipeline used by Wen and Rosé (2012), which is used as a baseline to compare our system performance. It consists of two main modules: medical event extractor and temporal resolver. However, there are a few differences from the baseline system. We do not use any filtering to remove sentences which do not contain mentions of self-reported events. Moreover, we use a different system for temporal expression extraction and normalization since the temporal resolved used by Wen and Rosé (2012) is not available for use. Because of these differences, we re-implement the baseline system in Wen and Rosé (2012) as described in section 6 to facilitate a fair comparison. After re-implementation, the only difference between the baseline and our system lies in the medical event extractor module. Instead of using manually designed keyword sets for extracting sentences containing medical events, we use a data-driven medical event extraction pipeline with minimal supervision. Fig 1 shows the architecture of our proposed system. We explain the modules for our proposed system in more detail in subsequent sections.

5 Modules for Proposed System

Our timeline construction system consists of two main modules: medical event extractor and temporal resolver.

5.1 Medical Event Extractor

We propose a pipeline for data-driven medical event detection using minimal supervision. Our pipeline comprises of three stages:

- **Medical Term Detection**
- **Medical Term Filtering**
- **Event Type Clustering**

In the following sections, we describe the algorithms used in these stages in more detail. We present evaluation results for each stage in section 6.

5.1.1 Medical Term Detection

In this stage, our aim is to select sentences which may contain mentions of a user's personal medical events. we use a simple rule to perform this selection: if a sentence contains a medical term, it is selected as a candidate sentence for the second stage of the pipeline. We experiment with two different methods for medical term detection.

[2] We use this subset because we could not get access to the full dataset used in Wen and Rosé (2012)

The first method uses ADEPT MacLean and Heer (2013), a medical term recognizer developed specifically for patient-authored text, to detect the presence of medical terms in sentences. All sentences containing at least one medical term, as detected by ADEPT, are chosen as candidate sentences. The second method is based on vocabulary difference between an online support group and a non-illness related group (VOCAB). We create term vocabularies from two groups on Ravelry: Knitters with Breast Cancer, a breast cancer support group and Beginners' Knit-Along, a non illness-related group. We then create a list of terms which occur at least once in Knitters with Breast Cancer, but do not appear at all in Beginners' Knit-Along. All terms in this list are now considered to be medical terms. Choosing two groups focusing on different interests from the same online community to detect medical terms, mitigates the problem of Ravelry-specific terms (such as Raveler, Ravatar etc.) being mistakenly included in the list. Using this term list, we perform candidate sentence extraction by choosing all sentences which contain at least one of the terms from the list.

Candidate sentences chosen by both methods contain a lot of spurious sentences, since many spurious terms are marked as medical terms by these methods. Hence, the next stage in our pipeline filters these candidate sentence sets.

5.1.2 Medical Term Filtering

In this stage, we filter out spurious terms to improve the quality of the candidate sentence set. We first discuss major sources of errors for both medical term detection methods and then discuss some strategies we use to mitigate these errors. These strategies are used to filter medical terms detected by both systems, which in turn filters candidate sentences selected by both.

We face one major issue while running the ADEPT system on our data. The system manages to correctly identify most important medical terms from the text, but it also marks several words used in non-medical contexts as medical terms. For example, in the sentence ''I must learn to speak more slowly than my brain thinks !'', the word ''brain'' is marked as a medical term, even though it not being used in a medical context. Performing such filtering is difficult, but we observe that when use a combination of terms from both methods, some of these errors get mitigated.

k	Vocab Size
1	28136
5	6585
10	2833
20	1197

Table 1: Massive decrease medical term vocabulary size with increasing value of k (the frequency limit for filtering)

While the vocabulary difference-based method does not fall into such context-based errors, it has its own drawbacks. Several terms in the list created via vocabulary difference are URLs, user names, email addresses and telephone numbers. We observe that such spurious terms are very infrequent. Hence, we perform filtering by removing all terms which occur with a frequency lower than k in the Knitters with Breast Cancer group, from our medical term list. We experiment with different values of k. Table 1 shows the massive reduction in medical term vocabulary with increasing value of k.

For further comparison, we evaluate the performance of both methods on candidate sentence extraction. These experiments and results are discussed further in section 6. Based on these results, we use a combination of both methods to perform medical event detection and filtering in the final system.

5.1.3 Event Type Clustering

In this stage, we use clustering to perform medical event type detection. We consider all sentences from the filtered set provided by the previous stage to be sentences containing mentions of medical events. This is an oversimplification since a sentence may contain a medical term which may not correspond to a medical event. For example, in the sentence ''My onc gave me the choice, saying she would rather ovrtreat than undertreat'', there are several medical terms (onc, overtreat, undertreat) but none of them are associated with medical events. However, we still perform clustering on the entire set, since such medical terms which do not correspond to medical events form a separate set of clusters which are later discarded. After clustering, we manually label each cluster with the medical event it corresponds to, and use these clusters as keyword sets to only retain sentences corresponding to each medical event. These sets of sentences for each medical event correspond to what Wen and Rosé (2012) call ”date sentences” and are used to extract the dates associated with these events.

We experiment with two methods for clustering. The first method uses Latent Dirichlet Allocation (Blei et al. (2003)) to cluster the candidate set of sentences. The use of this algorithm is motivated by the observation that people use similar expressions to describe the same medical events. However, as further discussed in section 6, we do not get good results using this algorithm.

The second method focuses on clustering medical terms instead of candidate sentences. We use a two-pass hierarchical clustering algorithm to cluster medical terms based on their Word2Vec (Mikolov et al. (2013)) embeddings. The word vectors used are pretrained on biomedical articles from Pubmed and PMC as well as English Wikipedia, in order to ensure enough domain specificity [3]. We also experiment with k-means clustering, but use agglomerative clustering for the final system due to better performance. In the first pass of agglomerative clustering, our main focus is on weeding out medical terms which are not linked to major cancer events. Hence, we run agglomerative clustering on all medical terms in this pass and manually inspect the produced clusters, discarding those which do not contain any terms corresponding to major cancer events. Thus, after the first pass, we are left with a list of terms, of which most are associated with major cancer events. This list of terms is then clustered during a second pass of agglomerative clustering. The final clusters produced by this pass are inspected and labeled with the cancer event that they correspond to. This method of clustering is able to identify better clusters, as discussed further in section 6. Hence, we use the keyword sets generated by this method for the final system.

5.2 Temporal Resolver

This module detects temporal expressions in every sentence, resolves those expressions to dates and then associates them with medical events. It has two phases: (1) temporal expression detection and resolution and (2) linking temporal expressions with events

5.2.1 Temporal Expression Detection and Resolution

We use HeidelTime (Strötgen and Gertz (2010)), a state-of-the-art temporal expression extractor and resolver to perform temporal expression detection and resolution on all candidate sentences extracted by the medical event extraction module. We run this system with the colloquial English setting, since our data comes from online support groups. Post timestamps are provided as document creation times.

5.2.2 Linking Temporal Expressions with Events

We use the rules of thumb proposed by Wen and Rosé (2012) to resolve temporal ambiguities, such as multiple temporal expressions occurring in a single sentence, for the baseline system. When multiple temporal expressions occur in the same sentence, the expression nearest to the event word in the sentence is chosen as the correct one. When an event is associated with multiple dates for the same user, we choose the most frequent date as the correct one.

6 Experimental Results and Evaluation

In this section, we first present our evaluation of each module for the proposed event detection pipeline. We then describe the performance achieved by our end-to-end system on the task of constructing cancer event timelines for users.

6.1 Evaluation of the Medical Term Detection Module

In this section, we evaluate the performance of two techniques (ADEPT-based term detection and vocabulary-based term detection) used in the medical term detection module. Since our aim is to replace the supervised sentence extraction phase in Wen and Rosé (2012) with our unsupervised pipeline while incurring minimal performance loss, we perform a comparative evaluation of this module. We use the sentence set extracted using manually defined keyword sets used by Wen and Rosé (2012) as our gold data. We measure performance by computing precision and recall of candidate sentence sets extracted by both medical term detection methods (ADEPT and VOCAB).Table 2 presents the precision, recall and F1 scores for these methods. We also present the scores for candidate sentence extraction using our vocabulary-based method before frequency-based filtering to

[3] These pretrained word vectors are provided by http://bio.nlplab.org/

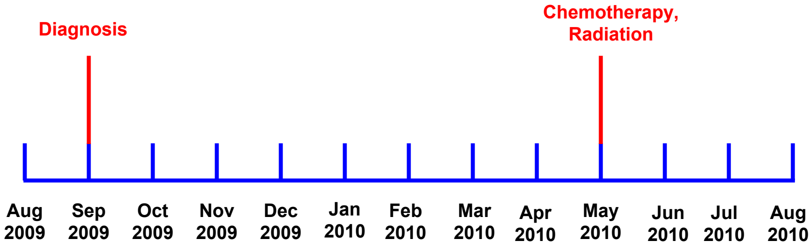

Figure 2: Sample cancer event timeline for a user constructed using events and dates extracted by our system

highlight the improvement achieved by filtering. As we can see from the table, each method has its own merits. ADEPT has extremely high precision but very poor recall, whereas our filtered term list improves significantly on precision while losing on recall. Hence, we combine both methods by only selecting sentences which contain terms marked as medical terms by both methods. As we can see from the table, this strategy works best, leading to an increase in precision without hurting recall [4]. This method is used in the final system.

6.2 Evaluation of the Clustering Module

We evaluate the performance of both methods (LDA and Word2Vec-based agglomerative clustering) used in the clustering module manually. We look at the words in each cluster generated by both clustering methods and label a cluster as corresponding to a certain cancer event, if most of the words in the cluster are associated with that event. For example, a cluster containing most diagnosis-related words (”diagnose”, ”diagnosis”, ”diagnosed” etc.) is labeled as the ”Diagnosis” event. On manual inspection of the clusters detected by LDA, we observe that only four important cancer events (chemotherapy, radiation, mastectomy and diagnosis) out of eight major events are identified. The main reason behind the poor performance of this algorithm is that the candidate sentences being clustered are very short and do not provide enough contextual information for the algorithm. However, we cannot perform clustering on the entire posts, since a single post may describe multiple events. Moreover, most cancer events co-occur with similar words and this further hampers LDA performance. On the other hand, a manual inspection of the clusters detected by Word2Vec-based agglomerative clustering detects six major cancer events (diagnosis, chemotherapy, radiation, reconstruction, metastasis, recurrence) very clearly. It also identifies a seventh event which is a mixture of words related to lumpectomy and mastectomy (it combines both these events into a single event). We present some keyword sets identified by this clustering algorithm for some cancer events below:

- **Chemotherapy:** chemotherapy, adjuvant, neoadjuvant, chemo, adriamycin, carboplatin, Taxol, herceptin, taxol, prednisone, Herceptin
- **Mastectomy/ Lumpectomy:** hysterectomy, lumpectomies, re-excision, mastectomy, Mastectomy, lumpectomy, mastectomies

As we can see from the above examples, these keyword sets are fairly coherent. [5]

6.3 End-to-End System Evaluation

To evaluate our end-to-end system, we test it on the user timeline construction task. As mentioned in section 3, we use a dataset consisting of all posts by a group of 50 users from breastcancer.org for this experiment. This dataset also contains date annotations for major cancer events for each user. However, this dataset is very small and only contains a total of 60 gold dates associated with cancer events, since dates pertaining to all cancer events for each user may not be available from their posting histories.

We compare the performance of our system with a re-implementation of the system described by Wen and Rosé (2012). We need to re-implement their system because we use different

[4] Increase in recall is observed because case-insensitive matching is used to find common terms selected by both ADEPT and VOCAB. This leads to the presence of some words selected only by one method in the final set. Such words however are case-variations of important words and must not be discarded

[5] It is difficult to peform a quantitative evaluation of the keyword sets since there is no gold standard

Method	Precision	Recall	F-Score
ADEPT	31.40	98.47	51.59
VOCAB (no filtering)	34.48	90.83	49.99
VOCAB (filtered)	47.46	84.17	60.45
ADEPT + VOCAB (filtered)	**50.32**	**99.03**	**66.73**

Table 2: Evaluation results for various methods used in the medical term detection module. For more details about these methods, refer to sections 5.1.1 and 5.1.2

System	Accuracy
(Wen and Rosé, 2012)	60
Our system	55.17

Table 3: Accuracy of date extraction for both systems on the cancer event timeline construction task

strategies for sentence filtering and temporal resolution, which can affect system performance. We use HeidelTime (Strötgen and Gertz, 2010) for temporal expression extraction and resolution. We also do not filter sentences which do not contain mentions of self-reported events. Hence, in order to facilitate a fair comparison between our system and (Wen and Rosé, 2012), we re-implement date sentence extraction (extraction of sentences containing medical events) as described in their paper, do not perform sentence filtering and use HeidelTime for temporal resolution. This version of the system is used as our baseline. We do this in order to ensure that the only difference between both systems lies in medical event extraction, which is the main focus of our work. We measure system performance based on accuracy, which is computed as the number of dates correctly extracted by the system divided by the number of dates present in the gold data. Table 3 presents the performance of both systems. From this comparison, we can see that our medical event extraction pipeline, in spite of being almost completely unsupervised, is able to achieve almost 92% of the accuracy obtained by the baseline system which uses supervised medical event extraction. However, the accuracy of both systems is not high enough to be used in practice.

Fig 2 shows a sample cancer event timeline created for a user. These cancer event timelines for users can be used to visualize patient disease trajectories. They can also be used to visualize links between important cancer events and user posting trends in online support groups by plotting the number of posts made by the user in each month on the same timeline and observing whether users tend to post more/ less during these events.

7 Error Analysis and Future Work

Since our dataset contains only 60 gold dates, our system misses only 3 dates as compared to the baseline system. Though the results of our current system are encouraging, deeper analysis of the errors made by the end-to-end system as well as the event clusters detected by our pipeline presents many shortcomings which should be addressed in future work.

Our system manages to detect six out of eight cancer events, but it is unable to distinguish between lumpectomy and mastectomy. Because of this, our system extracts the same date for both events. Though this is a small source of errors for the current system because the dataset is very small, this may turn out to be a large source of errors for bigger datasets. This error also shows that some medical events may be extremely similar and our current system might not be able to tease them apart. It would be desirable to come up with better clustering techniques which can make such fine distinctions.

Another source of errors for our system arises from the use of word vectors trained on PubMed and PMC articles. Since the word vectors are trained on biomedical data they contain a lot of medical terminology, but they do not contain appropriate word vectors for a lot of colloquial medical terms used in online support groups (eg: ”mets”, ”dx”). Hence such terms are not added to the correct cluster. For example, the words ”metastatic” and ”mets” appear in different clusters which is incorrect since they refer to the same event (metastasis). Transferring pre-trained word vectors from a biomedical corpus to data from an online support group can help mitigate this issue, which we plan to explore in the future.

An additional source of errors arises from the rules used to link temporal expressions to events. While we have rules which take care of the situation in which multiple temporal expressions may occur in the same sentence as the event, we ignore scenarios in which multiple event words may occur in a sentence with a single temporal expression. The current temporal rules will assign that expression to all events, which may be wrong in certain cases. For example, in the sentence ”Had lumpectomy in November 2000, but because margins were not clear, and another small tumor was found in the same breast, surgeon recommended modified radical mastectomy”, two cancer events (lumpectomy, mastectomy) are mentioned with only one temporal expression (November 2000). The temporal expression will be linked to both events according to the current resolution rules, whereas it should only be linked to lumpectomy. Moreover, sometimes the sentences may contain exactly one cancer event and one temporal expression. However, the temporal expression still does not refer to the cancer event. For example, in the sentence ”He died in Oct ’04, right after my bc diagnosis.”, the date October 2004 does not refer to the user’s diagnosis event. These issues with temporal resolution impact the performances of both our system and the baseline system. Improving strategies for linking events with temporal expressions should help in tackling these issues.

8 Conclusion

In this paper, we propose a novel data-driven pipeline for personal medical event extraction from social media using minimal supervision, which is able to achieve 92% of the performance achieved by a supervised baseline. The extracted medical events can be used to study and identify links between user participation on online support groups and important medical events in their lives. While the results of our current system pipeline for personal medical event extraction are encouraging, there is a lot of scope for further improvement.

Acknowledgments

This research was supported in parts by NSF IIS 1546393 and NHLBI R01 HL122639/. We would like to acknowledge Miaomiao Wen and Hyeju Jang for providing access to the data from Wen and Rosé (2012) and Khyathi Chandu for her feedback on earlier drafts of this paper. We would also like to thank the anonymous reviewers for their constructive feedback.

References

Hila Becker, Mor Naaman, and Luis Gravano. 2010. Learning similarity metrics for event identification in social media. In *Proceedings of the third ACM international conference on Web search and data mining*. ACM, pages 291–300.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3(Jan):993–1022.

Nathanael Chambers. 2013. Navytime: Event and time ordering from raw text. Technical report, DTIC Document.

Angel X Chang and Christopher D Manning. 2012. Sutime: A library for recognizing and normalizing time expressions. In *LREC*. volume 2012, pages 3735–3740.

Leon Derczynski, Jannik Strötgen, Diana Maynard, Mark A Greenwood, and Manuel Jung. 2016. Gatetime: Extraction of temporal expressions and event. In *10th Language Resources and Evaluation Conference*. European Language Resources Association (ELRA), pages 3702–3708.

Lifu Huang, T Cassidy, X Feng, H Ji, CR Voss, J Han, and A Sil. 2016. Liberal event extraction and event schema induction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-16).*

Jiwei Li and Claire Cardie. 2014. Timeline generation: Tracking individuals on twitter. In *Proceedings of the 23rd international conference on World wide web*. ACM, pages 643–652.

Jiwei Li, Alan Ritter, Claire Cardie, and Eduard H Hovy. 2014. Major life event extraction from twitter based on congratulations/condolences speech acts. In *EMNLP*. pages 1997–2007.

Diana Lynn MacLean and Jeffrey Heer. 2013. Identifying medical terms in patient-authored text: a crowdsourcing-based approach. *Journal of the American Medical Informatics Association* 20(6):1120–1127.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* .

Alan Ritter, Oren Etzioni, Sam Clark, et al. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pages 1104–1112.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*. ACM, pages 851–860.

Roser Saurí, Robert Knippen, Marc Verhagen, and James Pustejovsky. 2005. Evita: a robust event recognizer for qa systems. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 700–707.

Jannik Strötgen and Michael Gertz. 2010. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, pages 321–324.

Miaomiao Wen and Carolyn Penstein Rosé. 2012. Understanding participant behavior trajectories in online health support groups using automatic extraction methods. In *Proceedings of the 17th ACM international conference on Supporting group work*. ACM, pages 179–188.

Miaomiao Wen, Zeyu Zheng, Hyeju Jang, Guang Xiang, and Carolyn Penstein Rosé. 2013. Extracting events with informal temporal references in personal histories in online communities. In *ACL (2)*. pages 836–842.