TweeTime: A Minimally Supervised Method for Recognizing and Normalizing Time Expressions in Twitter

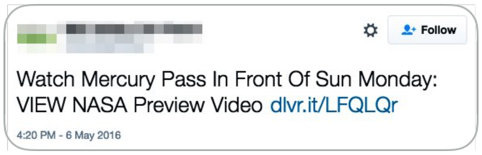
Jeniya Tabassum, Alan Ritter and Wei Xu
Computer Science and Engineering
Ohio State University
{bintejafar.1,ritter.1492,xu.1265}@osu.edu

Abstract

We describe TweeTIME, a temporal tagger for recognizing and normalizing time expressions in Twitter. Most previous work in social media analysis has to rely on temporal resolvers that are designed for well-edited text, and therefore suffer from reduced performance due to domain mismatch. We present a minimally supervised method that learns from large quantities of unlabeled data and requires no hand-engineered rules or hand-annotated training corpora. TweeTIME achieves 0.68 F1 score on the end-to-end task of resolving date expressions, outperforming a broad range of state-of-the-art systems.[1]

1 Introduction

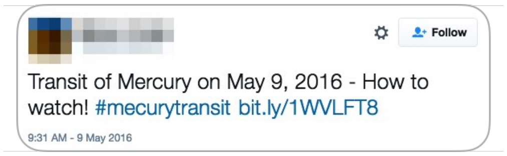
Temporal expressions are words or phrases that refer to dates, times or durations. Resolving time expressions is an important task in information extraction (IE) that enables downstream applications such as calendars or timelines of events (Derczynski and Gaizauskas, 2013; Do et al., 2012; Ritter et al., 2012; Ling and Weld, 2010), knowledge base population (Ji et al., 2011), information retrieval (Alonso et al., 2007), automatically scheduling meetings from email and more. Previous work in this area has applied rule-based systems (Mani and Wilson, 2000; Bethard, 2013b; Chambers, 2013) or supervised machine learning on small collections of hand-annotated news documents (Angeli et al., 2012; Lee et al., 2014).



Watch Mercury Pass In Front Of Sun Monday: VIEW NASA Preview Video dlvr.it/LFQLQr

4:20 PM - 6 May 2016

Figure 1: A tweet published on *Friday 5/6/2016* that contains the temporal expression *Monday* referring to the date of the event (*5/9/2016*), which a generic temporal tagger failed to resolve correctly.



Transit of Mercury on May 9, 2016 - How to watch! #mecurytransit bit.ly/1WVLFT8

9:31 AM - 9 May 2016

Figure 2: A tweet that contains a simple explicit time mention and an event (*Mercury, 5/9/2016*) that can be identified by an open-domain information extraction system.

Social media especially contains time-sensitive information and requires accurate temporal analysis, for example, for detecting real-time cybersecurity events (Ritter et al., 2015; Chang et al., 2016), disease outbreaks (Kanhabua et al., 2012) and extracting personal information (Schwartz et al., 2015). However, most work on social media simply uses generic temporal resolvers and therefore suffers from suboptimal performance. Recent work on temporal resolution focuses primarily on news articles and clinical texts (UzZaman et al., 2013; Bethard and Savova, 2016).

Resolving time expressions in social media is a non-trivial problem. Besides many spelling variations, time expressions are more likely to refer to future dates than in newswire. For the example in

[1] Our code and data are publicly available at https://github.com/jeniyat/TweeTime.

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 307–318,
Austin, Texas, November 1-5, 2016. ©2016 Association for Computational Linguistics

Figure 1, we need to recognize that *Monday* refers to the upcoming Monday and not the previous one to resolve to its correct normalized date (*5/9/2016*). We also need to identify that the word *Sun* is not referring to a Sunday in this context.

In this paper, we present a new minimally supervised approach to temporal resolution that requires no in-domain annotation or hand-crafted rules, instead learning from large quantities of unlabeled text in conjunction with a database of known events. Our approach is capable of learning robust time expression models adapted to the informal style of text found on social media.

For popular events, some related tweets (e.g. Figure 2) may contain explicit or other simple time mentions that can be captured by a generic temporal tagger. An open-domain information extraction system (Ritter et al., 2012) can then identify events (e.g. [*Mercury*, *5/9/2016*]) by aggregating those tweets. To automatically generate temporally annotated data for training, we make the following novel *distant supervision assumption*:[2]

> Tweets posted near the time of a known event that mention central entities are likely to contain time expressions that refer to the date of the event.

Based on this assumption, tweets that contain the same named entity (e.g. Figure 1) are heuristically labeled as training data. Each tweet is associated with multiple overlapping labels that indicate the day of the week, day of the month, whether the event is in the past or future and other time properties of the event date in relation to the tweet's creation date. In order to learn a tagger that can recognize temporal expressions at the word-level, we present a multiple-instance learning approach to model sentence and word-level tags jointly and handle overlapping labels. Using heuristically labeled data and the temporal tags predicted by the multiple-instance learning model as input, we then train a log-linear model that normalizes time expressions to calendar dates.

Building on top of the multiple-instance learning model, we further improve performance using a missing data model that addresses the problem of errors introduced during the heuristic labeling process. Our best model achieves a 0.68 F1 score when resolving date mentions in Twitter. This is a 17% increase over SUTime (Chang and Manning, 2012), outperforming other state-of-the-art time expression resolvers HeidelTime (Strötgen and Gertz, 2013), TempEX (Mani and Wilson, 2000) and UWTime (Lee et al., 2014) as well. Our approach also produces a confidence score that allows us to trade recall for precision. To the best of our knowledge, TweeTIME is the first time resolver designed specifically for social media data.[3] This is also the first time that distant supervision is successfully applied for end-to-end temporal recognition and normalization. Previous distant supervision approaches (Angeli et al., 2012; Angeli and Uszkoreit, 2013) only address the normalization problem, assuming gold time mentions are available at test time.

2 System Overview

Our TweeTIME system consists of two major components as shown in Figure 3:

1. A **Temporal Recognizer** which identifies time expressions (e.g. *Monday*) in English text and outputs 5 different temporal types (described in Table 1) indicating timeline direction, month of year, date of month, day of week or no temporal information (NA). It is realized as a multiple-instance learning model, and in an enhanced version, as a missing data model.

2. A **Temporal Normalizer** that takes a tweet with its creation time and temporal expressions tagged by the above step as input, and outputs their normalized forms (e.g. *Monday* → *5/9/2016*). It is a log-linear model that uses both lexical features and temporal tags.

To train these two models without corpora manually annotated with time expressions, we leverage a large database of known events as distant supervision. The event database is extracted automatically from Twitter using the open-domain IE system

[2] We focus on resolving dates, arguably the most important and frequent category of time expressions in social media data, and leave other phenomenon such as times and durations to traditional methods or future work.

[3] The closest work is HeidelTime's colloquial English version (Strötgen and Gertz, 2012) developed from annotated SMS data and slang dictionary. Our TweeTIME significantly outperforms on Twitter data.

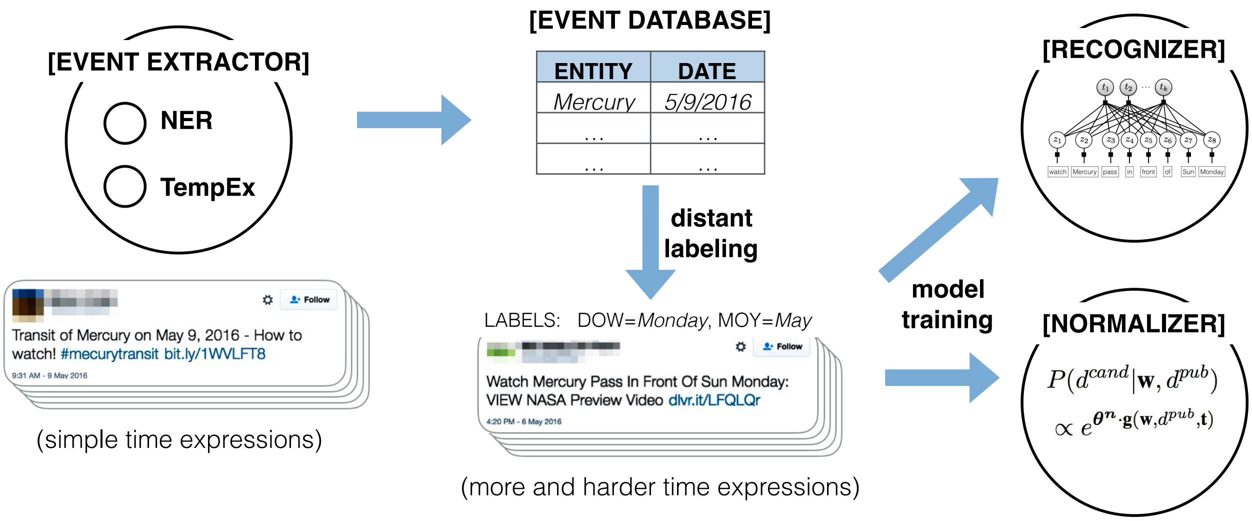

[EVENT EXTRACTOR]
NER
TempEx
Transit of Mercury on May 9, 2016 - How to watch! #mecurytransit bit.ly/1WVLFT8
9:31 AM - 9 May 2016
Follow
(simple time expressions)
[EVENT DATABASE]
ENTITY
DATE
Mercury
5/9/2016
...
distant labeling
LABELS: DOW=Monday, MOY=May
Watch Mercury Pass In Front Of Sun Monday: VIEW NASA Preview Video dlvr.it/LFQLQr
4:20 PM - 6 May 2016
Follow
(more and harder time expressions)
model training
[RECOGNIZER]
t_1 t_2 ... t_k
z_1 z_2 z_3 z_4 z_5 z_6 z_7 z_8
watch Mercury pass in front of Sun Monday
[NORMALIZER]
$P(d^{cand}|\mathbf{w}, d^{pub})$
$\propto e^{\boldsymbol{\theta_n} \cdot \mathbf{g}(\mathbf{w}, d^{pub}, \mathbf{t})}$

Figure 3: TweeTIME system diagram of model training.

Temporal Types	Possible Values (tags)
Timeline (TL)	$past, present, future$
Day of Week (DOW)	$Mon, Tue, \ldots, Sun$
Day of Month (DOM)	$1, 2, 3, \ldots, 31$
Month of Year (MOY)	$Jan, Feb, \ldots, Dec$
None (NA)	NA

Table 1: Our Temporal Recognizer can extract five different temporal types and assign one of their values to each word of a tweet.

proposed by Ritter et al. (2012). Each event consists of one or more named entities, in addition to the date on which the event takes place, for example [*Mercury*, *5/9/2016*]. Tweets are first processed by a Twitter named entity recognizer (Ritter et al., 2011), and a generic date resolver (Mani and Wilson, 2000). Events are then extracted based on the strength of association between each named entity and calendar date, as measured by a G^2 test on their co-occurrence counts. More details of the **Event Extractor** can be found in Section 5.1.

The following two sections describe the details of our **Temporal Recognizer** and **Temporal Normalizer** separately.

3 Distant Supervision for Recognizing Time Expressions

The goal of the recognizer is to predict the temporal tag of each word, given a sentence (or a tweet) $\mathbf{w} = w_1, \ldots, w_n$. We propose a multiple-instance learning model and a missing data model that are capable of learning word-level taggers given only sentence-level labels.

Our recognizer module in is built using a database of known events as *distant supervision*. We assume tweets published around the time of a known event that mention a central entity are also likely to contain time expressions referring to the event's date. For each event, such as [*Mercury*, *5/9/2016*], we gather all tweets that contain the central entity *Mercury* and are posted within 7 days of *5/9/2016*. We then label each tweet based on the event date in addition to the tweet's creation date. The sentence-level temporal tags for the tweet in Figure 1 are: TL=$future$, DOW=Mon, DOM=9, MOY=May.

3.1 Multiple-Instance Learning Temporal Tagging Model (MultiT)

Unlike supervised learning, where labeled instances are provided to the learner, in multiple instance learning scenarios (Dietterich et al., 1997), the learner is only provided with bags of instances labeled as either positive (where at least one instance is positive) or all negative. This is a close match to our problem setting, in which sentences are labeled with tags that should be assigned to one or more words.

We represent sentences and their labels using a graphical model that is divided into word-level and sentence-level variables (as shown in Figure 4). Unlike the standard supervised tagging prob-

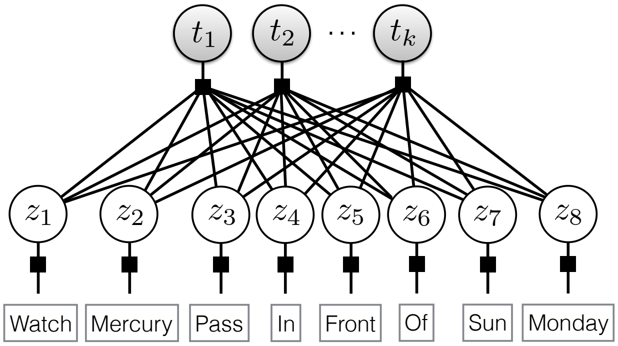


Figure 4: Multiple-Instance Learning Temporal Tagging Model – our approach to learn a word-level tagging model given only sentence-level labels. In this example a sentence-level variable $t_a = 1$ indicates the temporal tag DOW=Mon must be present and $t_b = 1$ indicates that the target date is in the future (TL=$future$). The multiple instance learning assumption implies that at least one word must be tagged with each of these present temporal tags. For example, ideally after training, the model will learn to assign z_8 to tag a and z_1 to tag b.

lem, we never directly observe the words' tags ($\mathbf{z} = z_1, \ldots, z_n$) during learning. Instead, they are latent and we only observe the date of an event mentioned in the text, from which we derive sentence-level binary variables $\mathbf{t} = t_1, \ldots, t_k$ corresponding to temporal tags for the sentence. Following previous work on multiple-instance learning (Hoffmann et al., 2011a; Xu et al., 2014), we model the connection between sentence-level labels and word-level tags using a set of deterministic-OR factors ϕ^{sent}.

The overall conditional probability of our model is defined as:

$$\begin{aligned} &P(\mathbf{t}, \mathbf{z}|\mathbf{w}; \boldsymbol{\theta^r}) \\ &= \frac{1}{Z}\prod_{i=1}^{k}\phi^{sent}(t_i, \mathbf{z}) \times \prod_{j=1}^{n}\phi^{word}(z_j, w_j) \\ &= \frac{1}{Z}\prod_{i=1}^{k}\phi^{sent}(t_i, \mathbf{z}) \times \prod_{j=1}^{n}e^{\boldsymbol{\theta^r}\cdot\mathbf{f}(z_j, w_j)} \end{aligned} \tag{1}$$

where $\mathbf{f}(z_j, w_j)$ is a feature vector and

$$\phi^{sent}(t_i, \mathbf{z}) = \begin{cases} 1 & \text{if } t_i = true \wedge \exists j : z_j = i \\ 1 & \text{if } t_i = false \wedge \forall j : z_j \neq i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

We include a standard set of tagging features that includes word shape and identity in addition to prefixes and suffixes. To learn parameters $\boldsymbol{\theta^r}$ of the Temporal Tagger, we maximize the likelihood of the sentence-level heuristic labels conditioned on observed words over all tweets in the training corpus. Given a training instance $\mathbf{w}$ with label $\mathbf{t}$, the gradient of the conditional log-likelihood with respect to the parameters is:

$$\begin{aligned} \nabla P(\mathbf{t}|\mathbf{w}) = &\sum_{\mathbf{z}} P(\mathbf{z}|\mathbf{w}, \mathbf{t}; \boldsymbol{\theta^r}) \cdot \mathbf{f}(\mathbf{z}, \mathbf{w}) \\ &- \sum_{\mathbf{t},\mathbf{z}} P(\mathbf{t}, \mathbf{z}|\mathbf{w}; \boldsymbol{\theta^r}) \cdot \mathbf{f}(\mathbf{z}, \mathbf{w}) \end{aligned} \tag{3}$$

This gradient is the difference of two conditional expectations over the feature vector $\mathbf{f}$: a "clamped" expectation that is conditioned on the observed words and tags ($\mathbf{w}$, $\mathbf{t}$) and a "free" expectation that is only conditioned on the words in the text, $\mathbf{w}$, and ignores the sentence-level labels. To make the inference tractable, we use a Viterbi approximation that replaces the expectations with maximization. Because each sentence corresponds to more than one temporal tag, the maximization of the "clamped" maximization is somewhat challenging to compute. We use the approximate inference algorithm of Hoffmann et al. (2011a), that views inference as a weighted set cover problem, with worst case running time $(|T| \cdot |W|)$, where $|T|$ is the number of all possible temporal tag values and $|W|$ is the number of words in a sentence.

3.2 Missing Data Temporal Tagging Model (MiDaT)

While the multiple-instance learning assumption works well much of the time, it can easily be violated – there are many tweets that mention entities involved in an event but that never explicitly mention its date.

The missing data modeling approach to weakly supervised learning proposed by Ritter et. al. (2013) addresses this problem by relaxing the hard constraints of deterministic-OR factors, such as those described above, as soft constraints. Our missing-data model for weakly supervised tagging splits the sentence-level variables, t into two parts : m which represents whether a temporal tag is mentioned by at least one word of the tweet, and t' which represents whether a temporal tag can be derived from the event date. A set of pairwise potentials $\psi(m_j, t'_j)$ are introduced that encourage (but don't strictly require) agreement between m_j and t'_j, that is:

$$\psi(m_j, t'_j) = \begin{cases} \alpha_p, \text{ if } t'_j \neq m_j \\ \alpha_r, \text{ if } t'_j = m_j \end{cases} \quad (4)$$

Here, α_p (Penalty), and α_r (Reward) are parameters for the MiDaT model. α_p is the penalty for extracting a temporal tag that is not related to the event-date and α_r is the reward for extracting a tag that matches the date.

During learning, if the local classifier is very confident, it is possible for a word to be labeled with a tag that is not derived from the event-date, and also for a sentence-level tag to be ignored, although either case will be penalized by the agreement potentials, $\psi(m_j, t'_j)$, in the global objective. We use a local-search approach to inference that was empirically demonstrated to nearly always yield exact solutions by Ritter et. al. (2013).

4 A Log-Linear Model for Normalizing Time Expressions

The Temporal Normalizer is built using a log-linear model which takes the tags **t** produced by the Temporal Recognizer as input and outputs one or more dates mentioned in a tweet. We formulate date normalization as a binary classification problem: given a tweet **w** published on date d^{pub}, we consider 22 candidate target dates (**w**, d_l^{cand}) such that $d_l^{cand} = d^{pub} + l$, where $l = -10, \ldots, -1, 0, +1, \ldots, +10$, limiting the possible date references that are considered within 10 days before or after the tweet creation date, in addition to $d_l^{cand} = null$ (the tweet does not mention a date). [4] While our basic approach has the limitation, that it is only able to predict dates within ±10 days of the target date, we found that in practice the majority of date references on social media fall within this window. Our approach is also able to score dates outside this range that are generated by traditional approaches to resolving time expressions, as described in Section 5.3.3.

The normalizer is similarly trained using the event database as distant supervision. The probability that a tweet mentions a candidate date is estimated using a log-linear model:

$$P(d^{cand}|\mathbf{w}, d^{pub}) \propto e^{\boldsymbol{\theta^n} \cdot \mathbf{g}(\mathbf{w}, d^{pub}, \mathbf{t})} \quad (5)$$

where $\boldsymbol{\theta^n}$ and **g** are the parameter and feature vector respectively in the Temporal Normalizer. For every tweet and candidate date pair (**w**, d_l^{cand}), we extract the following set of features:

Temporal Tag Features that indicate whether the candidate date agrees with the temporal tags extracted by the Temporal Recognizer. Three cases can happen here: The recognizer can extract a tag that can not be derived from the candidate date; The recognizer can miss a tag derived from the candidate date; The recognizer can extract a tag that is derived from the candidate date.

Lexical Features that include two types of binary features from the tweet: 1) **Word Tag** features consist of conjunctions of words in the tweet and tags associated with the candidate date. We remove URLs, stop words and punctuation; 2) **Word POS** features that are the same as above, but include conjunctions of POS tags, words and temporal tags derived from the candidate date.

Time Difference Features are numerical features that indicate the distance between the creation date and the candidate date. They include difference of day ranges form -10 to 10 and the difference of week ranges from -2 to 2.

5 Experiments

In the following sub-sections we present experimental results on learning to resolve time expressions in Twitter using minimal supervision. We start by describing our dataset, and proceed to present our results, including a large-scale evaluation on heuristically-labeled data and an evaluation comparing against human judgements.

5.1 Data Collection

We collected around 120 million tweets posted in a one year window starting from April 2011 to May 2012. These tweets were automatically annotated with named entities, POS tags and TempEx dates (Ritter et al., 2011).

[4] Although the temporal recognizer is trained with tweets from ±7 days around the event date, we found that extending the candidate date range to ±10 days for the temporal normalizer increased the performance of TweeTIME in the dev set.

From this automatically-annotated corpus we extract the top $10,000$ events and their corresponding dates using the G^2 test, which measures the strength of association between an entity and date using the log-likelihood ratio between a model in which the entity is conditioned on the date and a model of independence (Ritter et al., 2012). Events extracted using this approach then simply consist of the highest-scoring entity-date pairs, for example [*Mercury, 5/9/2016*].

After automatically extracting the database of events, we next gather all tweets that mention an entity from the list that are also written within ± 7 days of the event. These tweets and the dates of the known events serve as labeled examples that are likely to mention a known date.

We also include a set of pseudo-negative examples, that are unlikely to refer to any event, by gathering a random sample of tweets that do not mention any of the top $10,000$ events and where TempEx does not extract any date.

5.2 Large-Scale Heuristic Evaluation

We first evaluate our tagging model, by testing how well it can predict the heuristically generated labels. As noted in previous work on distant supervision (Mintz et al., 2009a), this type of evaluation usually under-estimates precision, however it provides us with a useful intrinsic measure of performance.

In order to provide even coverage of months in the training and test set, we divide the twitter corpus into 3 subsets based on the mod-5 week of each tweet's creation date. To train system we use tweets that are created in $1st$, $2nd$ or $3rd$ weeks. To tune parameters of the MiDaT model we used tweets from $5th$ weeks, and to evaluate the performance of the trained model we used tweets from $4th$ weeks.

	Precision	**Recall**	**F-value**
MultiT	0.61	0.21	0.32
MiDaT	0.67	0.31	0.42

Table 2: Performance comparison of MultiT and MiDaT at predicting heuristically generated tags on the dev set.

The performance of the MiDaT model varies with the penalty and reward parameters. To find a (near) optimal setting of the values we performed a grid search on the dev set and found that a penalty of -25 and reward of 500 works best. A comparison of MultiT and MiDaT's performance at predicting heuristically generated labels is shown in Table 2.

The word level tags predicted by the temporal recognizer are used as the input to the temporal normalizer, which predicts the referenced date from each tweet. The overall system's performance at predicting event dates on the automatically generated test set, compared against SUTime, is shown in Table 3.

	System	**Prec.**	**Recall**	**F-value**
dev set	TweeTIME	0.93	0.69	0.79
	SUTime	0.89	0.64	0.75
test set	TweeTIME	0.97	0.94	0.96
	SUTime	0.85	0.75	0.80

Table 3: Performance comparison of TweeTIME and SUTime at predicting heuristically labeled normalized dates.

5.3 Evaluation Against Human Judgements

In addition to automatically evaluating our tagger on a large corpus of heuristically-labeled tweets, we also evaluate the performance of our tagging and date-resolution models on a random sample of tweets taken from a much later time period, that were manually annotated by the authors.

5.3.1 Word-Level Tags

To evaluate the performance of the MiDaT-tagger we randomly selected 50 tweets and labeled each word with its corresponding tag. Against this hand annotated test set, MiDaT achieves Precision=0.54, Recall=0.45 and F-value=0.49. A few examples of word-level tags predicted by MiDaT are shown in Table 4. We found that because the tags are learned as latent variables inferred by our model, they sometimes don't line up exactly with our intuitions but still provide useful predictions, for example in Table 4, *Christmas* is labeled with the tag MOY=dec.

5.3.2 End-to-end Date Resolution

To evaluate the final performance of our system and compare against existing state-of-the art time resolvers, we randomly sampled 250 tweets from 2014-2016 and manually annotated them with normalized dates; note that this is a separate date range from our weakly-labeled training data which is taken from 2011-2012. We use 50 tweets as a development set and the remaining 200 as a final test set.

Tweets and their corresponding word tags (word^{tag})
Im^{NA} hella^{NA} excited^{future} for^{NA} tomorrow^{future}
Kick^{NA} off^{NA} the^{NA} New^{future} Year^{future} Right^{NA} @^{NA} #ClubLacura^{NA} #FRIDAY^{fri} !^{NA} HOSTED^{NA} BY^{NA} [[^{NA} DC^{NA} Young^{NA} Fly^{NA}]]^{NA}
@OxfordTownHall^{NA} Thks^{NA} for^{NA} a^{NA} top^{NA} night^{NA} at^{NA} our^{NA} Christmas^{dec} party^{NA} on^{NA} Fri!^{fri} Compliments^{NA} to^{NA} chef!^{NA} (Rose^{NA} melon^{NA} *cantaloupe*^{NA} :)^{NA}
Im^{NA} proud^{NA} to^{NA} say^{NA} that^{NA} I^{NA} breathed^{past} the^{NA} same^{NA} air^{NA} as^{NA} Harry^{NA} on^{NA} March^{mar} 21,^{21} 2015.^{NA} #KCA^{NA} #Vote1DUK^{NA}
C'mon^{present} let's^{present} jack^{NA} Tonight^{present} will^{NA} be^{present} a^{NA} night^{NA} to^{NA} remember.^{NA}

Table 4: Example MiDaT tagging output on the test set.

	Precision	**Recall**	**F-value**
TweeTIME	0.61	0.81	0.70
- Day Diff.	0.46	0.72	0.56
- Lexical&POS	0.48	0.80	0.60
- Week Diff.	0.49	0.85	0.62
- Lexical	0.50	0.88	0.64
- Temporal Tag	0.57	0.83	0.68

Table 5: Feature ablation of the Temporal Resolver by removing each individual feature group from the full set.

	System	**Prec.**	**Recall**	**F-value**
dev set	TweeTIME	**0.61**	0.81	**0.70**
	TweeTIME+SU	**0.67**	**0.83**	**0.74**
	SUTime	0.51	**0.86**	0.64
	TempEx	0.58	0.64	0.61
	HeidelTime	0.57	0.63	0.60
	UWTime	0.49	0.57	0.53
test set	TweeTIME	**0.58**	**0.70**	**0.63**
	TweeTIME+SU	**0.62**	**0.76**	**0.68**
	SUTime	0.54	0.64	0.58
	TempEx	0.56	0.58	0.57
	HeidelTime	0.43	0.52	0.47
	UWTime	0.39	0.50	0.44

Table 6: Performance comparison of TweeTIME against state-of-the-art temporal taggers. TweeTIME+SU uses our proposed approach to system combination, re-scoring output from SUTime using extracted features and learned parameters from TweeTIME.

We experimented with different feature sets on the development data. Feature ablation experiments are presented in Table 5.

The final performance of our system, compared against a range of state-of-the-art time resolvers is presented in Table 6. We see that TweeTIME outperforms SUTime, Tempex, HeidelTime (using its COLLOQUIAL mode, which is designed for SMS text) and UWTime. Brief descriptions of each system can be found in Section 6.

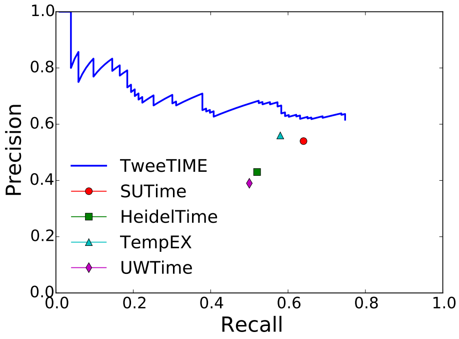


Figure 5: Precision and recall at resolving time expressions compared against human judgements. TweeTIME achieves higher precision at comparable recall than other state-of-the-art systems.

5.3.3 System Combination with SUTime

As our basic TweeTIME system is designed to predict dates within ±10 days of the creation date, it fails when a tweet refers to a date outside this range. To overcome this limitation we append the date predicted by SUTime in the list of candidate days. We then re-rank SUTime's predictions using our log-linear model, and include its output as a predicted date if the confidence of our normalizer is sufficiently high.

5.3.4 Error Analysis

We manually examined the system outputs and found 7 typical categories of errors (see examples in Table 7):

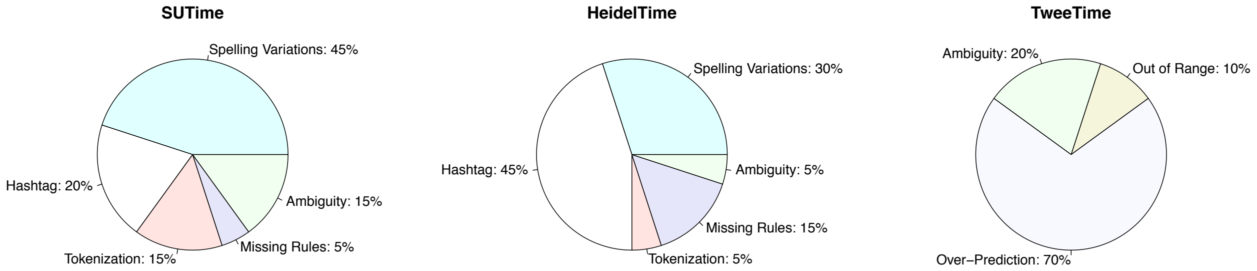


Figure 6: Error analyses for different temporal resolvers

Spelling Variation: Twitter users are very creative in their use of spelling and abbreviations. For example, a large number of variations of the word *tomorrow* can be found in tweets, including *2morrow*, *2mrw*, *tmrw*, *2mrow* and so on. Previous temporal resolvers often fail in these cases, while TweeTIME significantly reduces such errors.

Ambiguity: In many cases, temporal words like *Friday* in the tweet *Is it Friday yet?* may not refer to any specific event or date, but are often predicted incorrectly. Also included in this category are cases where the future and past are confused. For example, predicting the past Friday, when it is actually the coming Friday.

Missing Rule: Cases where specific temporal keywords, such as *April Fools*, are not covered by the rule-based systems.

Tokenization: Traditional systems tend to be very sensitive to incorrect tokenization and have trouble to handle expressions such as *9th-december*, *May 9,2015* or *Jan1*. For the following Tweet:

> JUST IN Delhi high court asks state government to submit data on changes in pollution level since #OddEven rule came into effect on *Jan1*

TweeTIME is able to correctly extract *01/01/2016*, whereas HeidelTime, SUTime, TempEX and UWTime all failed to extract any dates.

Hashtag: Hashtags can carry temporal information, for example, #September11. Only our system that is adapted to social media can resolve these cases.

Out of Range: TweeTIME only predicts dates within 10 days before or after the tweet. Time expressions referring to dates outside this range will not be predicted correctly. System combination with SUTime (Section 5.3.3) only partially addressed this problem.

Over-Prediction: Unlike rule-based systems, TweeTIME has a tendency to over-predict when there is no explicit time expression in the tweets, possibly because of the presence of present tense verbs. Such mistakes could also happen in some past tense verbs.

Because TweeTIME resolves time expressions using a very different approach compared to traditional methods, its distribution of errors is quite distinct, as illustrated in Figure 6.

6 Related Work

Temporal Resolvers primarily utilize either rule-based or probabilistic approaches. Notable rule-based systems such as TempEx (Mani and Wilson, 2000), SUTime (Chang and Manning, 2012) and HeidelTime (Strötgen and Gertz, 2013) provide particularly competitive performance compared to the state-of-the-art machine learning methods. Probabilistic approaches use supervised classifiers trained on in-domain annotated data (Kolomiyets and Moens, 2010; Bethard, 2013a; Filannino et al., 2013) or hybrid with hand-engineered rules (UzZaman and Allen, 2010; Lee et al., 2014). UWTime (Lee et al., 2014) is one of the most recent and competitive systems and uses Combinatory Categorial Grammar (CCG).

Although the recent research challenge TempEval (UzZaman et al., 2013; Bethard and Savova, 2016) offers an evaluation in the clinical domain besides newswire, most participants used the provided annotated corpus to train supervised models in addition to employing hand-coded rules. Previous work on adapting temporal taggers primarily focus on scaling up to more languages. HeidelTime was extended to multilingual (Strötgen and Gertz, 2015), colloquial (SMS) and scientific texts (Strötgen and Gertz, 2012) using dictionaries and additional in-domain

Error Category	Tweet	Gold Date	Predicted Date
Spelling	I cant believe *tmrw* is *fri*..the week flys by	2015-03-06	None (SUTime, HeidelTime)
Ambiguity	RT @Iyaimkatie: Is it Friday yet?????	None	2015-12-04 (TweeTime, SUTime, HeidelTime)
Missing Rule	#49ers #sanfrancisco 49ers fans should be oh so wary of *April Fools* pranks	2015-04-01	None (HeidelTime)
Tokenization	100000 - still waiting for that reply from *9th-december* lmao. you're pretty funny and chill	2015-12-09	None (SUTime, HeidelTime)
Hashtag	RT @arianatotally: Who listening to the *#SATURDAY* #Night w/ @AlexAngelo?I'm loving it.	2015-04-11	None (SUTime, HeidelTime)
Out of Range	RT @460km: In memory of Constable Christine Diotte @rcmpgrcpolice EOW: *March 12, 2002* #HeroesInLife #HerosEnVie	2002-03-12	2015-03-12 (TweeTime)
Over-Prediction	RT @tinatbh: January 2015: this will be my year December 2015: maybe not.	None	2015-12-08 (TweeTime)

Table 7: Representative Examples of System (SUTime, HeidelTime, TweeTIME) Errors

annotated data. One existing work used distant supervision (Angeli et al., 2012; Angeli and Uszkoreit, 2013), but for normalization only, assuming gold time mentions as input. They used an EM-style bootstrapping approach and a CKY parser.

Distant Supervision has recently become popular in natural language processing. Much of the work has focused on the task of relation extraction (Craven and Kumlien, 1999; Bunescu and Mooney, 2007; Mintz et al., 2009b; Riedel et al., 2010; Hoffmann et al., 2011b; Nguyen and Moschitti, 2011; Surdeanu et al., 2012; Xu et al., 2013; Ritter et al., 2013; Angeli et al., 2014). Recent work also shows exciting results on extracting named entities (Ritter et al., 2011; Plank et al., 2014), emotions (Purver and Battersby, 2012), sentiment (Marchetti-Bowick and Chambers, 2012), as well as finding evidence in medical publications (Wallace et al., 2016). Our work is closely related to the joint word-sentence model that exploits multiple-instance learning for paraphrase identification (Xu et al., 2014) in Twitter.

7 Conclusions

In this paper, we showed how to learn time resolvers from large amounts of unlabeled text, using a database of known events as distant supervision. We presented a method for learning a word-level temporal tagging models from tweets that are heuristically labeled with only sentence-level labels. This approach was further extended to account for the case of missing tags, or temporal properties that are not explicitly mentioned in the text of a tweet. These temporal tags were then combined with a variety of other features in a novel date-resolver that predicts normalized dates referenced in a Tweet. By learning from large quantities of in-domain data, we were able to achieve 0.68 F1 score on the end-to-end time normalization task for social media data, significantly outperforming SUTime, TempEx, HeidelTime and UWTime on this challenging dataset for time normalization.

Acknowledgments

We would like to thank the anonymous reviewers for helpful feedback on a previous draft. This material is based upon work supported by the National Science Foundation under Grant No. IIS-1464128. Alan Ritter is supported by the Office of the Director of National Intelligence (ODNI) and the Intelligence Advanced Research Projects Activity (IARPA) via the Air Force Research Laboratory (AFRL) contract number FA8750-16-C-0114. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, AFRL, or the U.S. Government.

References

Omar Alonso, Michael Gertz, and Ricardo Baeza-Yates. 2007. On the value of temporal information in information retrieval. In *ACM SIGIR Forum*, volume 41, pages 35–41. ACM.

Gabor Angeli and Jakob Uszkoreit. 2013. Language-independent discriminative parsing of temporal expressions. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*.

Gabor Angeli, Christopher D Manning, and Daniel Jurafsky. 2012. Parsing time: Learning to interpret time expressions. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Gabor Angeli, Julie Tibshirani, Jean Wu, and Christopher D Manning. 2014. Combining distant and partial supervision for relation extraction. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Steven Bethard and Guergana Savova. 2016. SemEval-2016 Task 12: Clinical TempEval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval)*.

Steven Bethard. 2013a. ClearTK-TimeML: A minimalist approach to TempEval 2013. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval)*.

Steven Bethard. 2013b. A synchronous context free grammar for time normalization. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Razvan C. Bunescu and Raymond J. Mooney. 2007. Learning to extract relations from the Web using minimal supervision. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Nathanael Chambers. 2013. NavyTime: Event and time ordering from raw text. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*.

Angel X Chang and Christopher D Manning. 2012. SUTime: A library for recognizing and normalizing time expressions. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*.

Ching-Yun Chang, Zhiyang Teng, and Yue Zhang. 2016. Expectation-regulated neural model for event mention extraction. *Proccedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Technologies (NAACL)*.

Mark Craven and Johan Kumlien. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology (ISMB)*.

Leon Derczynski and Robert J Gaizauskas. 2013. Temporal signals help label temporal relations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*.

Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 89(1).

Quang Xuan Do, Wei Lu, and Dan Roth. 2012. Joint inference for event timeline construction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP)*.

Michele Filannino, Gavin Brown, and Goran Nenadic. 2013. ManTIME: Temporal expression identification and normalization in the TempEval-3 challenge. In *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval)*.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011a. Knowledge-based weak supervision for information extraction of overlapping relations. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL)*.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke S. Zettlemoyer, and Daniel S. Weld. 2011b. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Heng Ji, Ralph Grishman, Hoa Trang Dang, Kira Griffitt, and Joe Ellis. 2011. Overview of the tac 2011 knowledge base population track. In *Proceedings of the Fourth Text Analysis Conference (TAC)*.

Nattiya Kanhabua, Sara Romano, Avaré Stewart, and Wolfgang Nejdl. 2012. Supporting temporal analytics for health-related events in microblogs. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*.

Oleksandr Kolomiyets and Marie-Francine Moens. 2010. KUL: Recognition and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*.

Kenton Lee, Yoav Artzi, Jesse Dodge, and Luke Zettlemoyer. 2014. Context-dependent semantic parsing for time expressions. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Xiao Ling and Daniel S Weld. 2010. Temporal information extraction. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI)*.

Inderjeet Mani and George Wilson. 2000. Robust temporal processing of news. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics (ACL)*.

Micol Marchetti-Bowick and Nathanael Chambers. 2012. Learning for microblogs with distant supervision: Political forecasting with Twitter. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009a. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the Association of Computational Linguistics and the International Joint Conference on Natural Language Processing (ACL-IJCNLP)*.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009b. Distant supervision for relation extraction without labeled data. In *Proceedigns of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (ACL)*.

Truc-Vien T. Nguyen and Alessandro Moschitti. 2011. End-to-end relation extraction using distant supervision from external semantic repositories. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Barbara Plank, Dirk Hovy, Ryan McDonald, and Anders Søgaard. 2014. Adapting taggers to twitter with not-so-distant supervision. pages 1783–1792.

Matthew Purver and Stuart Battersby. 2012. Experimenting with distant supervision for emotion classification. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedigns of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*.

Alan Ritter, Mausam, Sam Clark, and Oren Etzioni. 2011. Named entity recognition in Tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Alan Ritter, Mausam, Oren Etzioni, and Sam Clark. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD)*.

Alan Ritter, Luke Zettlemoyer, Mausam, and Oren Etzioni. 2013. Modeling missing data in distant supervision for information extraction. *Transactions of the Association for Computational Linguistics (TACL)*, 1:367–378.

Alan Ritter, Evan Wright, William Casey, and Tom Mitchell. 2015. Weakly supervised extraction of computer security events from Twitter. In *Proceedings of the 24th International Conference on World Wide Web (WWW)*.

H Andrew Schwartz, Greg Park, Maarten Sap, Evan Weingarten, Johannes Eichstaedt, Margaret Kern, Jonah Berger, Martin Seligman, and Lyle Ungar. 2015. Extracting human temporal orientation in Facebook language. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Jannik Strötgen and Michael Gertz. 2012. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Jannik Strötgen and Michael Gertz. 2015. A baseline temporal tagger for all languages. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Naushad UzZaman and James F Allen. 2010. TRIPS and TRIOS system for Tempeval-2: Extracting temporal information from text. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*.

Naushad UzZaman, Hector Llorens, James Allen, Leon Derczynski, Marc Verhagen, and James Pustejovsky. 2013. SemEval-2013 Task 1: TEMPEVAL-3: Evaluating time expressions, events, and temporal relations. In *Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval)*.

Byron C Wallace, Joël Kuiper, Aakash Sharma, Mingxi Brian Zhu, and Iain J Marshall. 2016. Extracting PICO sentences from clinical trial reports using supervised distant supervision. *Journal of Machine Learning Research (JMLR)*.

Wei Xu, Raphael Hoffmann, Zhao Le, and Ralph Grishman. 2013. Filling knowledge base gaps for distant supervision of relation extraction. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Wei Xu, Alan Ritter, Chris Callison-Burch, William B. Dolan, and Yangfeng Ji. 2014. Extracting lexically divergent paraphrases from Twitter. *Transactions of the Association for Computational Linguistics (TACL)*, 2(1).