Learning to Extract Events from Knowledge Base Revisions

Alexander Konovalov
Ohio State University
konovalov.2@osu.edu

Benjamin Strauss
Ohio State University
strauss.105@osu.edu

Alan Ritter
Ohio State University
ritter.1492@osu.edu

Brendan O'Connor
University of Massachusetts, Amherst
brenocon@cs.umass.edu

ABSTRACT

Broad-coverage knowledge bases (KBs) such as Wikipedia, Freebase, Microsoft's Satori and Google's Knowledge Graph contain structured data describing real-world entities. These data sources have become increasingly important for a wide range of intelligent systems: from information retrieval and question answering, to Facebook's Graph Search, IBM's Watson, and more. Previous work on learning to populate knowledge bases from text has, for the most part, made the simplifying assumption that facts remain constant over time. But this is inaccurate—we live in a rapidly changing world. Knowledge should not be viewed as a *static snapshot*, but instead a rapidly evolving set of facts that must change as the world changes.

In this paper we demonstrate the feasibility of accurately identifying *entity-transition-events*, from real-time news and social media text streams, that drive changes to a knowledge base. We use Wikipedia's edit history as distant supervision to learn event extractors, and evaluate the extractors based on their ability to predict online updates. Our weakly supervised event extractors are able to predict 10 KB revisions per month at 0.8 precision. By lowering our confidence threshold, we can suggest 34.3 correct edits per month at 0.4 precision. 64% of predicted edits were detected before they were added to Wikipedia. The average lead time of our forecasted knowledge revisions over Wikipedia's editors is 40 days, demonstrating the utility of our method for suggesting edits that can be quickly verified and added to the knowledge graph.[1]

1. INTRODUCTION

An entity's properties are frequently affected by events taking place in the world. For example, an *election* event can change the LEADER property of a country, or a *wedding* can change the SPOUSE of a person. When these events take place, Wikipedia editors frequently update infobox[2] properties of the affected entities.

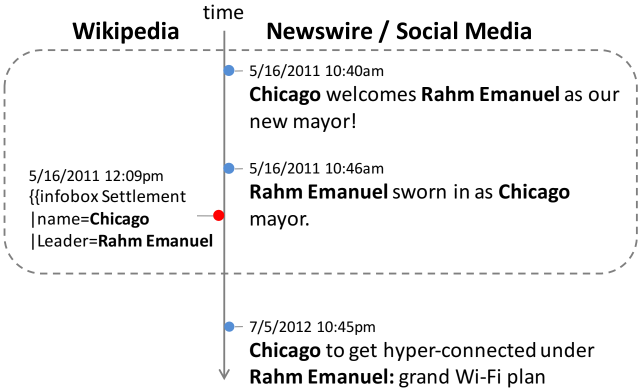


Figure 1: Aligning Wikipedia's infobox edits to events mentioned in text.

In this paper we show that it is possible to train event extractors for streaming text using the revision history of Wikipedia's structured infobox data as a distant source of supervision. Our weakly supervised event extraction models are capable of automatically recommending revisions to the knowledge graph in realtime; an evaluation of our proposed edits using workers from Amazon's Mechanical Turk demonstrates that our models automatically suggest, on average, 34.3 correct edits per month, 64% of which were predicted before human knowledge base contributors to Wikipedia.

Previous work has examined extracting event records from text [12, 13] usually within narrow domains—for example, in international relations event extraction, researchers have sought to extract a historical database of interactions from a news archive, by developing rule-based [29], supervised [4] and unsupervised [20] learning methods.

But *distant supervision*[3] is a successful alternative approach to populating knowledge graphs, by aligning sentences to records in a pre-existing database. It scales up to a wide variety of relations, and can exploit redundancies within large text corpora. Previous work [1, 18, 11, 31, 27, 34, 5, 28] has mostly focused on extracting relationships that remain relatively stable over time. A small amount of related work has tackled the problem of extracting events using distant supervision. So far, work has been limited to narrow domains such as plane crashes [23], live performances [3]; and in a non-KB context, temporally scoped polling data can constitute distant supervision to learn a social media sentiment classifier [17].

[1] Code and data will be made publicly available upon publication.
[2] http://en.wikipedia.org/wiki/Infobox

[3] Also known as weak supervision.

WWW 2017, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4913-0/17/04.
http://dx.doi.org/10.1145/3038912.3052646

Part of the challenge in extending distant supervision to events in broader domains is the reliance of weakly supervised learning methods on redundancy [6] - while many sentences on the web are likely to mention context independent relationships, such as the headquarters of a company, most events are only mentioned in a handful of news articles; this lack of redundancy has made learning event extractors from unaligned structured data very challenging.

In the meantime, social networking websites such as Twitter have become an important complementary source to traditionally studied corpora such as newswire and the web. When important events take place, many users independently turn to microblogs to share information resulting in a large number of redundant messages describing each significant event and providing an opportunity to collect large amounts of training data for weakly supervised event extraction [25, 26, 32].

In contrast to prior work, our method can leverage Wikipedia's vast revision history as a source of distant (weak) supervision for learning to extract events from text.

Contributions

Our contributions in this paper are the following:

- We propose the use of knowledge base revisions as a novel source of distant supervision for learning to extract events from large-scale streaming text data.
- We present techniques for identifying reliable infobox edits corresponding to an entity's change of state.
- We introduce a dataset for this task which enables learning news and social media extractors from Wikipedia infobox revisions.
- We experimentally demonstrate the feasibility of our method for learning to extract events associated with Wikipedia infobox edits. After verifying our highest-confidence predictions using workers from Amazon's Mechanical Turk, we show that we can generate on average 34.3 edits per month, often beating human knowledge base contributors in terms of recall and lead time.

2. LEARNING TO EXTRACT EVENTS FROM KNOWLEDGE BASE REVISIONS

Our approach to predicting knowledge-base edits is to learn extractors for events that alter properties of knowledge-base entities, by leveraging the revision history of Wikipedia's semi-structured data as weak supervision.

To predict a revision changing e_1's attribute to e_2 at an arbitrary time t, we use evidence gathered from tweets written before t that mention both entities (e_1 and e_2). These entities correspond to the Wikipedia page title and new value for the attribute, respectively.

We model the probability of an edit adding e_2 as a new value of attribute r for entity e_1 at time t using a log-linear model:

$$P_r(E_r|e_1, e_2, t) \propto \exp(\theta_r \cdot f(e_1, e_2, t))$$

Features, $f(e_1, e_2, t)$, are computed from text written before t taken from a set of timestamped documents. Parameters for each relation, θ_r, are chosen to maximize the likelihood of an observed set of Wikipedia edits conditioned on an archive of timestamped documents. An overview of our process for aligning observed edits to text for feature extraction is presented in Figure 1. Features used for event prediction consist of sequences of words surrounding the entities, for example: "e_2 *sworn in as* e_1 *mayor*" or "e_1 *marries...girlfriend* e_2". Full details on our approach to alignment and feature extraction are presented in Sections 3 and 4.

After training our models, we apply named entity taggers to raw news and social media streams, and consider every pair of entities mentioned together within the same sentence as a candidate entity-transition event adding e_2 as a new value for each attribute r of entity e_1.

As a concrete dataset, we experiment with the revision history of Wikipedia's infoboxes, in addition to two sources of realtime text: Twitter and newswire. In the following sections we describe each of these datasets and our NLP preprocessing pipeline (including part of speech tagging and named entity recognition), followed by our approach to gathering observed knowledge revisions to use as supervision for training edit prediction models.

2.1 Wikipedia's Infobox Revision History

When important events take place, Wikipeda's editors often quickly update attributes of relevant entities. However not all Wikipedia revisions are made in response to current events - some simply fill in missing information that has long been public. There is also a great deal of vandalism and edits whose purpose is simply improving visual presentation of the page.

To cope with these challenges, we propose 2 simple but effective heuristics for identifying semantic edits in response to real-word events and filter out those due to vandalism and other causes:

Novel Attribute Values: We only consider edits that introduce a previously unseen value for an entity's attribute. For instance if *Rob Ford* is added as a value of the LEADER attribute of *Toronto*, then removed and later added again, this is likely due to re-formatting of the infobox's appearance or vandalism.

Persistent Edits: We define a particular modification of an infobox entry to be *persistent* if there are no edits which remove this new value within ten days after initial modification. We filter out non-persistent modifications to eliminate spurious edits associated with vandalism.

These heuristics help to filter out most of the noise, however they are not perfect - they do admit some vandalism and non-semantic edits. Most of these are naturally removed later during the matching step, however, when we align edits to textual event mentions.

We applied these two filtering steps to the data-set extracted by Alfonseca et al. [2], which contains $38,979,871$ infobox attribute updates for $1,845,172$ different entities, and is freely available for download. The data includes all edits to Wikipedia's infoboxes from June 2004 until January 2012.

For this work we selected a set of 6 infobox attributes whose changes correspond to certain well-defined events happening in the world:

CURRENTTEAM**:** An athlete's current team.

LEADERNAME**:** The leader of a geopolitical entity (e.g. mayor, president, prime minister).

STATEREPRESENTATIVE**:** The U.S. state represented by a politician.

SPOUSE**:** A person's husband or wife.

PREDECESSOR**:** The previous person to hold the same political office or other position.

DEATHPLACE**:** A person's place of death.

2.2 Twitter Data

We used the Twitter Streaming API to continuously collect a sample of all public tweets, archiving this stream from September 2008 through December 2012. (For most of the time period this consists of approximately 10% of all public tweets.) To conduct the analyses in this paper, we additionally downsampled messages to a maximum of 1 million per day, to control for changes to the overall volume over time.

We filtered to English messages using *langid.py* [16], and tagged each tweet with named entities and parts of speech using an in-domain Twitter NLP pipeline [24]. This process resulted in 239,156,419 messages containing at least one named entity from the 1st of September 2008 until the 1st of January 2012.

2.3 Annotated Gigaword

To explore differences between knowledge-revision events mentioned in Twitter and traditional news media, we used the Annotated Gigaword v.5 dataset [19] to represent newswire as an additional source of realtime text. Gigaword is a large static corpus of English news articles comparable in size to our Twitter corpus. In total it contains $4,111,240$ documents with dates ranging from April 1994 until December 2010. Annotated gigaword includes state-of-the-art syntactic parses automatically extracted named entities, and a variety of other linguistic annotations.

We separated each news document into individual sentences (analogous to tweets). This allowed us to use practically the same downstream extraction pipeline for both Twitter and Gigaword. We used document dates to generate timestamps for each sentence.

2.4 Matching Tweets and News Sentences to Wikipedia Edits

Significant prior work has investigated the problem of linking textual mentions to entities in a knowledge base [7, 10, 15, 14]. To tackle the challenge of aligning knowledge revisions to text, we take a simple and efficient approach that uses surface-form matching and readily-available alias dictionaries [30]. This approach results in accurate alignment between text and knowledge base revisions when combined with additional context from pairs of entities and restriction to a time window near the date of the edit.

For this work we selected a set of 6 infobox attributes that have a high number of matching tweets, and whose changes correspond to certain well-defined events happening in the world: CURRENTTEAM, LEADERNAME, STATEREPRESENTATIVE, SPOUSE, PREDECESSOR, and DEATHPLACE.

2.5 Training, Development, and Test Periods

We separated each dataset by time, using earlier data for training edit prediction models, and testing on data created afterward. For Twitter, we used the following time periods:

Training: September 1st 2008 - June 1st 2011

Test: June 5th 2011 - January 1st 2012

Similarly, we separated the Gigaword corpus into the following training and test periods:

Training: June 1st 2004 - June 1st 2009

Test: June 5th 2009 - January 1st 2011

3. ALIGNING KNOWLEDGE BASE REVISIONS TO TEXT

In practice, we never directly observe actual *entity transition events*, $E_r(e_1, e_2)$. Instead, we observe only a corresponding Wikipedia edit some time afterward, and tweets or newswire sentences written in response to the event.

The time difference between an edit and a corresponding event mentioned in text could be a matter of minutes or days, and the text could be written either before or after the edit depending on a variety of factors. Typically however, we found that Wikipedia edits occur within several days of the news first being reported in text.

Attribute	#Tweets (training)	#Tweets (test)
Predecessor	6,034	503
DeathPlace	11239	1713
State	27,696	2,015
LeaderName	36,041	5,736
CurrentTeam	137,717	17,025
Spouse	74,665	14,254
Randomly sampled	1,319,403	5,766,526

Table 1: Number of tweets aligned to infobox edits, for training and automatic evaluation.

Attribute	#Sent. (training)	#Sent. (test)
Predecessor	90,617	22,193
DeathPlace	65,450	51,328
State	148,275	37,924
LeaderName	665,364	187,980
CurrentTeam	200,612	76,078
Spouse	41,524	8,102
Randomly Sampled	11,467,027	3,118,503

Table 2: Number of newswire sentences and documents matching infobox edits in the Gigaword corpus.

Based on this intuition, we separate all tweets containing a pair of entities associated with an edit (e_1, e_2) into three disjoint sets, based on their relative time difference. Specific values for parameters were manually tuned on a separate development set:

- T_{aligned} contains text that is well-aligned with an edit and most likely to mention the event. Tweets or news sentences in this set happen no more than 10 days before the edit and no later than 3 days afterward.
- $T_{\text{unaligned}}$ contains text that mentions the entity pair but was written significantly earlier or later than the edit and is unlikely to reference the event. Tweets or news sentences in this set are written more than 50 days before the edit or later than 3 days afterward.
- The remaining tweets contain text written before the event but it is not clear whether it refers to the event. We do not use this data for training purposes. Tweets or news sentences in this set appear between 10 and 50 days before the edit.

Examples of aligned tweets and Wikipedia revisions are presented in Table 3, and the number of tweets and news sentences matching Wikipedia revisions in our datasets for training and automatic evaluation are presented in Tables 1 and 2.

In addition to sentences matching Wikipedia infoboxes, we also include a sample that are not matched to *any* edit, as negative training examples. We will refer to this portion of the data as T_{random}.

Due to memory constraints we reduced the number of Twitter messages and Gigaword sentences in T_{random} by randomly subsampling down to 2% of the original number of entity pairs.

3.1 Rolling Prediction Window

Since any individual tweet or newswire sentence may give an incorrect cue, during prediction, we extract features from all data within a time window of 24 hours preceding a prediction at time t.

Wikipedia Edit				Twitter	
e_1	attribute	e_2	Date	Date	sample tweet
Brazil	LEADER	Dilma Rousseff	1/1/2011	11/1/2010	Congratulations to **Dilma Rousseff** , first female President of **Brazil** ! This is epic !
Japan	LEADER	Naoto Kan	7/1/2010	6/6/2010	Barack Obama telephones **Japan** 's new leader **Naoto Kan** to pledge co-operation
Australia	LEADER	Julia Gillard	11/7/2010	6/24/2010	First woman Prime Minister in **Australia** .. **Julia Gillard** .
Toronto	LEADER	Rob Ford	12/1/2010	10/26/2010	Anti-Bike , Anti-Transit , Anti-Green **Rob Ford** Elected Mayor Of **Toronto**
New Zealand	LEADER	John Key	11/18/2008	11/10/2008	**New Zealand** 's prime minister-elect **John Key** has arrived in the ..
Japan	LEADER	Taro Aso	9/24/2008	9/24/2008	Outspoken conservative **Taro Aso** took power as **Japan** 's new Prime Mi ..
Prince William	SPOUSE	Kate Middleton	4/29/2011	11/17/2010	RT @BBCBreaking : Clarence House has said **Prince William** is to marry **Kate Middleton** next year .
Katy Perry	SPOUSE	Russell Brand	10/23/2010	10/24/2010	**Katy Perry** Marries **Russell Brand** in True Bollywood Fashion
Anna Paquin	SPOUSE	Stephen Moyer	8/22/2010	8/22/2010	True Blood ' stars **Anna Paquin** , **Stephen Moyer** wed in California
Javier Bardem	SPOUSE	Penelope Cruz	7/13/2010	7/15/2010	**Penelope Cruz** &**Javier Bardem** got married this month too ?! So many marriages this July !
LeBron James	TEAM	Miami Heat	7/10/2010	7/9/2010	**LeBron James** to Play With **Miami Heat** Next Season
Carmelo Anthony	TEAM	new york knicks	2/22/2011	2/22/2011	**Carmelo Anthony** traded to **New York Knicks**
Carl Crawford	TEAM	Red Sox	12/11/2010	12/9/2010	**Carl Crawford** + Adrian Gonzalez + **Red Sox** = WS Championship .

Table 3: Examples of knowledge revisions and corresponding event mentions in text. A sample tweet is shown which was written near the same date as the edit, and mentions both the page title (e_1) and new attribute value (e_2), as detected by an in-domain named entity recognizer.

We make a prediction every time a new tweet is created that mentions one or more entity pairs and extract features from all tweets written within the prediction window mentioning the entity pair.

3.2 Training Event Prediction Models

Our first Event Prediction system employs the following distant supervision assumption: **tweets that are written near the time of a knowledge graph revision are likely to mention an event that causes the change in state**. To implement this assumption, we label samples in T_{aligned} as positive.

Next, we assume that all tweets in $T_{\text{unaligned}} \cup T_{\text{random}}$ are negative examples of revisions for the target attribute. This set consists largely of mentions of the relation, which are not associated with any change of state to the knowledge graph, for example: "e_1*'s husband* e_2", or "e_1 and e_2 *welcome a new baby*" for the SPOUSE attribute.

3.3 Relation Extraction Baseline

As a baseline we consider a relation extraction system trained in the same way as previous work on distant supervision [18], which ignores temporal information and is simply trained to predict whether a relation holds between a pair of entities. The Relation Extraction baseline assumes every matching training sample in T_{matched} is positive, and all randomly sampled tweets in T_{random} are pseudo-negatives. At test time, the system is presented with features extracted from tweets in the same window as the Event Prediction system.

Note that the only difference between event and relation prediction is during training. They are both supplied with exactly the same features extracted from the prediction windows at test time.

4. FEATURES

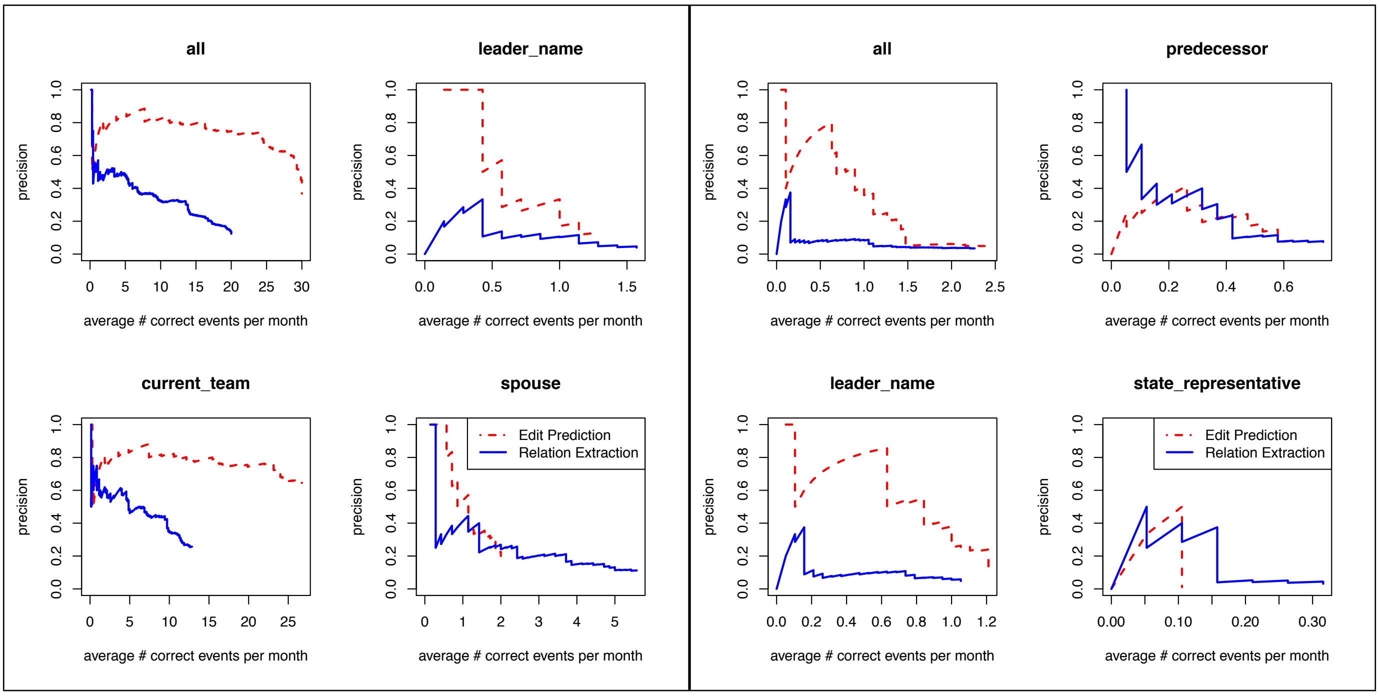
We use the following templates to extract features, $f(e_1, e_2, t)$, for an entity pair (e_1, e_2) from all samples written within the prediction window for use in forecasting a knowledge revision at time t:

- A window of $k = 1, 2, 3$ words to the left and right of each entity and 6 words at most between the two entities (if there are more than 6 words, some are replaced with an ellipsis "..."). Examples include: *"e_1 ,...married e_2", "wedding : e_2 and e_1"*.

- The same features as above but with any non-noun or non-verb replaced by its respective part-of-speech tag. Examples include: *"e_2 CC e_1 tied DT", "e_1 agrees TO JJ deal IN e_2"*.

More examples of high-weight features are presented in Table 4

We removed all entity pairs with fewer than 5 corresponding tweets or newswire sentences in the training time period. We also removed features that fire on fewer than 5 entity pairs, resulting in a reduction in the number of features from 65,225,828 to 69,960.



(a) Twitter

(b) Gigaword

Figure 2: Human evaluation of automatically extracted knowledge revisions from (a) Twitter and (b) Gigaword. Each graph plots the precision vs. average number of events extracted per month. Although we annotated all 6 relations for each dataset, here we are only showing relations that produced a non-trivial number of correct extractions. No useful extractions were produced from the Twitter data for the PREDECESSOR and STATE relations, while only very low precision extractions were generated from Gigaword for SPOUSE and CURRENTTEAM. Note that the x-axis varies between graphs as many more events are extracted in certain cases (e.g. CURRENTTEAM).

Attribute	Feature	Weight
SPOUSE	e_2 marries e_1	1.05
SPOUSE	e_1 and e_2 wedding	0.67
SPOUSE	e_1 and e_2 are getting married	0.63
CURRENTTEAM	e_2 sign e_1	1.55
CURRENTTEAM	e_1 to...the e_2	1.36
CURRENTTEAM	e_1 traded to e_2	1.25
LEADERNAME	e_1 's...elect e_2	0.90
LEADERNAME	e_1 elected e_2	0.54
LEADERNAME	e_1 's new...prime minister e_2	0.52

Table 4: Examples of high-weight features learned for Twitter.

5. EVALUATION

We performed 2 evaluations of the feasibility of predicting knowledge-base revisions from news and social media: a large scale evaluation of how well we can predict actual edits to Wikipedia, in addition to a human evaluation of predicted edits using Amazon's Mechanical Turk.

5.1 Large-Scale Evaluation Against Wikipedia Edits

For the purposes of automatic evaluation on the testing period, we used T_{aligned} as positive examples, $T_{\text{unaligned}}$ and a subset of T_{random} as negative examples. We randomly sampled the subset of T_{random} to correspond to 90% of the testing data. This is similar to our label assignment for training of the event prediction system. F_1-scores predicting real edits in Wikipedia are presented in Tables 5 and 6 for Twitter and Gigaword respectively.

5.2 Evaluation with Human Judgements

Automatic evaluation provides us with a useful measure of our system's performance during development. It is also interesting to see how many infobox edits generated by Wikipedia contributors we can predict. However, it underestimates both precision and recall because many of our automatically generated edits add new knowledge that wasn't updated by Wikipedians. Our collection of edits also contains some noise in the form of vandalism and stylistic edits. This motivates the need for human annotators to inspect our highest confidence predictions to directly evaluate their quality.

To this end, we conducted human evaluation using Amazon's crowd-sourcing platform, Mechanical Turk. Workers were shown the latest tweet in each prediction window and asked if the entity-transition-event was correctly extracted. A separate question template was written specifically for each attribute. Workers were asked to select one of the following options: Yes, No, or Not Sure. Each human intelligence task (HIT) consisted of a collection of 10 questions. An example question template for the CURRENTTEAM relation is presented in Figure 3. For quality control each HIT was completed by seven different workers. No subsampling was performed on negative data for the human evaluation, so this emulates a realistic testing scenario where edits are predicted from both Twitter and Gigaword.

We separated output from each system into month-long bins and selected the top prediction for each entity pair for each month that

Attribute	Event P/R/F_1			Relation P/R/F_1		
CurrentTeam	0.68	0.43	0.53	0.31	0.26	0.28
LeaderName	1.00	0.33	0.50	0.09	0.21	0.12
Spouse	0.35	0.21	0.27	0.14	0.60	0.22
StateRepresentative	0.004	0.50	0.01	8e-4	0.50	0.002
Predecessor	0.01	0.25	0.02	0.06	0.25	0.09
DeathPlace	0.61	0.40	0.49	0.70	0.58	0.63

Table 5: Automatic evaluation on the Twitter dataset (Maximum F_1 scores and corresponding precision and recall evaluated at the same threshold level).

Attribute	Event P/R/F_1			Relation P/R/F_1		
CurrentTeam	0.02	0.11	0.03	0.02	0.45	0.03
LeaderName	0.47	0.19	0.27	0.04	0.12	0.05
Spouse	0.12	0.05	0.08	0.03	0.15	0.05
StateRepresentative	0.78	0.30	0.43	0.58	0.20	0.29
Predecessor	0.28	0.22	0.24	0.15	0.20	0.17
DeathPlace	0.38	0.45	0.41	0.38	0.32	0.35

Table 6: Automatic evaluation on the Gigaword dataset (Maximum F_1 scores and corresponding precision and recall evaluated at the same threshold level).

was not already added to Wikipedia or predicted by our system before that month. To reduce the number of duplicate predictions we used Levenshtein Distance to compare entity pairs at this step; entity pairs with at most 3 character-level edits were considered the same.

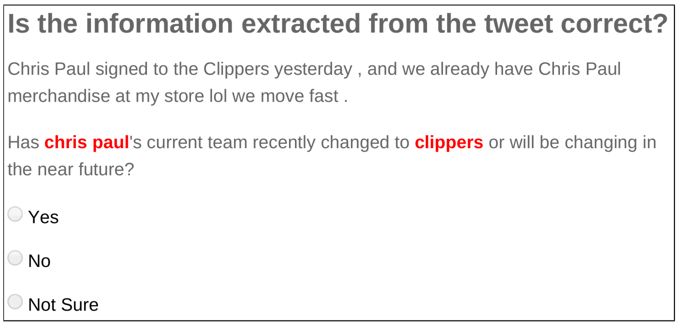
We annotated at least the 10 highest confidence predictions for each system for each month in the test set. Because some relations produce significantly more predictions than others (e.g. athletes are frequently traded between teams, but changes in a country's leadership are relatively infrequent), we annotated up to 50 predictions per month such that the estimated probability of an edit is above a threshold, $P(\text{edit}|e_1, e_2, t) > 0.3$. This resulted in a total of 4,415 predicted infobox edits over the test time period that were presented to the Mechanical Turk workers for evaluation. Each proposed edit was evaluated by 7 workers; we consider an edit correctly extracted if the majority of workers labeled it as such (e.g. at least 4).

To determine inter-annotator agreement of the Turkers we used the Fleiss Kappa metric [8] treating the "Not Sure" annotations as "No". Our workers had Fleiss Kappa agreement of 0.64 on the Twitter data and 0.30 on news sentences from Gigaword.

We hypothesize two factors that could explain this lower agreement rate on Gigaword: (1) longer, more complex sentences that are more difficult for the Mechanical Turk workers to read and (2) a larger proportion of referring expressions and missing context from the full article.

Attr	No Matching Edit	After Edit	Before Edit
CurrentTeam	11.714	7.429	4.429
LeaderName	0.143	0.429	0.429
Predecessor	0.0	0.0	0.0
Spouse	1.143	0.286	0.571
StateRepresentative	0.0	0.0	0
DeathPlace	2.143	0.571	0.286
Total	15.143	8.715	5.715

Table 7: Average number of correct Twitter predictions per month that were generated before and after matching Wikipedia edits. The higher proportion of non-matching edits in Twitter is mostly due to a mix of rumors and missing Wikipedia entries.



Is the information extracted from the tweet correct?

Chris Paul signed to the Clippers yesterday , and we already have Chris Paul merchandise at my store lol we move fast .

Has **chris paul**'s current team recently changed to **clippers** or will be changing in the near future?

○ Yes

○ No

○ Not Sure

Figure 3: Example question template presented to Mechanical Turk users for evaluation of the CurrentTeam relation.

Attr	No Matching Edit	After Edit	Before Edit
CurrentTeam	0.0	0.158	0.053
LeaderName	0.053	0.737	0.316
Predecessor	0.0	0.368	0.053
Spouse	0.053	0.053	0.158
StateRepresentative	0.0	0.105	0.0
DeathPlace	0.158	2.316	0.158
Total	0.264	3.737	0.738

Table 8: Average number of correct Gigaword predictions per month that were generated before and after matching Wikipedia edits.

6. DISCUSSION

Both the human evaluation presented in Figure 2 and automatic evaluation in Table 5 paint a consistent picture of our performance across our 7 test attributes. Twitter and newswire appear complementary in predicting entity-transition-events. Neither data source produces better performance for all of the relations. For instance from Twitter, we are able to propose on average about 30 correct CurrentTeam edits per month with precision 0.64 using our edit prediction models, whereas only a handful of correct predictions were made during the entire test period of the Gigaword corpus. Using Gigaword we are able to extract roughly 1 correct edit to the LeaderName attribute every other month with high precision, whereas from Twitter we were only able to extract 1 edit every several months.

6.1 Comparison to Wikipedia's Human Editors

Can we automatically predict edits to Wikipedia's infoboxes from news and social media before human knowledge-base contributors? In addition to exploring the accuracy of our extracted events we compared the time of our proposed edits with timestamps of real knowledge-base updates from Wikipedia's revision history.

As illustrated in Tables 7 and 8, a significant proportion of our predicted edits are generated before any Wikipedia editor contributed the same information. From Twitter we are able to extract 5.7 edits per month for the 6 relations considered in this study before Wikipedia editors contributed the same relation. In addition, 15 events per month were either missed by Wikipedia's editors or were rumors; from Gigaword we predicted 0.7 edits every month ahead of the Wikipedia editors. In this work we do not attempt to distinguish between rumors and genuine news about an event, although this task has been addressed to some extent in prior work [22, 37].

7. RELATED WORK

Past work has explored retrospectively scoping beginning and end times of facts in a knowledge base [33]. Previous work has also investigated populating Wikipedia's infoboxes from the text of associated articles [36, 35]. For example automatically updating an infobox's BirthDate or BirthPlace attribute when associated information is added to the text of an article. In contrast we extract information from realtime text streams as soon as public knowledge of an event becomes available. The Knowledge Base Acceleration track at TREC [9] shares our emphasis on tracking and extracting structured information as soon as an event is developing, however the KBA track uses manually annotated data while we exploit KB revision history using weak supervision.

Also relevant is work on first story detection using events detected on social media in combination with realtime information on Wikipedia page views to identify high quality breaking events [21]. Our work, in contrast, is focused on extracting structured events in the form of predicted edits to a knowledge base.

8. CONCLUSIONS

We have considered the problem of predicting revisions to a knowledge base using events extracted from text. Our approach leverages the KB's revision history as distant supervision for learning to extract entity-transition-events that alter properties of knowledge base entities. We empirically demonstrated the feasibility of training a variety of event extractors for both Twitter and newswire using the revision history of Wikipedia's infoboxes proposing on average 34.3 edits per month, 64% of which were either predicted before human knowledge base contributors to Wikipedia, or were missed by Wikipedia's editors. On average, our Twitter-based forecasts beat the Wikipedia editors by 40 days.

Acknowledgments

We would like to thank Wei Xu and 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. Alexander Konovalov, Benjamin Strauss and Alan Ritter are 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.

9. REFERENCES

[1] E. Agichtein and L. Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the fifth ACM conference on Digital libraries*, 2000.

[2] E. Alfonseca, G. Garrido, J.-Y. Delort, and A. Peñas. Whad: Wikipedia historical attributes data. *Language Resources and Evaluation*, 47(4):1163–1190, 2013.

[3] E. Benson, A. Haghighi, and R. Barzilay. Event discovery in social media feeds. In *ACL*, 2011.

[4] E. Boschee, P. Natarajan, and R. Weischedel. Automatic extraction of events from open source text for predictive forecasting. *Handbook of Computational Approaches to Counterterrorism*, page 51, 2013.

[5] K.-W. Chang, W.-t. Yih, B. Yang, and C. Meek. Typed tensor decomposition of knowledge bases for relation extraction. In *EMNLP*, pages 1568–1579, 2014.

[6] D. Downey, O. Etzioni, and S. Soderland. A probabilistic model of redundancy in information extraction. Technical report, IJCAI, 2005.

[7] Y. Fang and M.-W. Chang. Entity linking on microblogs with spatial and temporal signals. *Transactions of the Association for Computational Linguistics*, 2014.

[8] J. L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[9] J. R. Frank, M. Kleiman-Weiner, D. A. Roberts, E. Voorhees, and I. Soboroff. Evaluating stream filtering for entity profile updates in trec 2012, 2013, and 2014 (kba track overview, notebook paper). Technical report, DTIC Document, 2014.

[10] S. Guo, M.-W. Chang, and E. Kiciman. To link or not to link? a study on end-to-end tweet entity linking. In *HLT-NAACL*, 2013.

[11] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld. Knowledge-based weak supervision for information extraction of overlapping relations. In *ACL*, 2011.

[12] H. Ji and R. Grishman. Refining event extraction through cross-document inference. In *ACL*, pages 254–262, 2008.

[13] Q. Li, H. Ji, and L. Huang. Joint event extraction via structured prediction with global features. In *ACL (1)*, pages 73–82, 2013.

[14] X. Ling, S. Singh, and D. S. Weld. Design challenges for entity linking. *Transactions of the Association for Computational Linguistics*, 2015.

[15] X. Liu, Y. Li, H. Wu, M. Zhou, F. Wei, and Y. Lu. Entity linking for tweets. In *ACL (1)*, 2013.

[16] M. Lui and T. Baldwin. langid. py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 system demonstrations*, 2012.

[17] M. Marchetti-Bowick and N. Chambers. 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*, pages 603–612. Association for Computational Linguistics, 2012.

[18] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of ACL-IJCNLP 2009*, 2009.

[19] C. Napoles, M. Gormley, and B. Van Durme. Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 95–100. Association for Computational Linguistics, 2012.

[20] B. O'Connor, B. Stewart, and N. A. Smith. Learning to extract international relations from political context. In *Proceedings of ACL*, 2013.

[21] M. Osborne, S. Petrovic, R. McCreadie, C. Macdonald, and I. Ounis. Bieber no more: First story detection using twitter and wikipedia. In *SIGIR 2012 Workshop on Time-aware Information Access*, 2012.

[22] V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei. Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1589–1599. Association for Computational Linguistics, 2011.

[23] K. Reschke, M. Jankowiak, M. Surdeanu, C. D. Manning, and D. Jurafsky. Event extraction using distant supervision. In *Proceedings of LREC 2014*, 2014.

[24] A. Ritter, S. Clark, Mausam, and O. Etzioni. Named entity recognition in tweets: An experimental study. *EMNLP*, 2011.

[25] A. Ritter, O. Etzioni, S. Clark, et al. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1104–1112. ACM, 2012.

[26] A. Ritter, E. Wright, W. Casey, and T. Mitchell. Weakly supervised extraction of computer security events from twitter. In *Proceedings of the 24th International Conference on World Wide Web*.

[27] A. Ritter, L. Zettlemoyer, Mausam, and O. Etzioni. Modeling missing data in distant supervision for information extraction. *Transactions of the Association for Computational Linguistics*, 1:367–378, 2013.

[28] T. Rocktaschel, S. Singh, and S. Riedel. Injecting logical background knowledge into embeddings for relation extraction. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2015.

[29] P. A. Schrodt. Automated coding of international event data using sparse parsing techniques. *International Studies Association Conference*, 2001.

[30] S. Singh, A. Subramanya, F. Pereira, and A. McCallum. Wikilinks: A large-scale cross-document coreference corpus labeled via links to wikipedia. *University of Massachusetts, Amherst, Tech. Rep. UM-CS-2012-015*, 2012.

[31] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. D. Manning. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics, 2012.

[32] J. Tabassum, A. Ritter, and W. Xu. Tweetime : A minimally supervised method for recognizing and normalizing time expressions in twitter. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016.

[33] P. P. Talukdar, D. Wijaya, and T. Mitchell. Coupled temporal scoping of relational facts. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 73–82. ACM, 2012.

[34] R. West, E. Gabrilovich, K. Murphy, S. Sun, R. Gupta, and D. Lin. Knowledge base completion via search-based question answering. In *WWW*, 2014.

[35] D. T. Wijaya, N. Nakashole, and T. Mitchell. "a spousal relation begins with a deletion of engage and ends with an addition of divorce": Learning state changing verbs from wikipedia revision history. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2015.

[36] F. Wu and D. S. Weld. Autonomously semantifying wikipedia. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 41–50. ACM, 2007.

[37] Z. Zhao, P. Resnick, and Q. Mei. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1395–1405. International World Wide Web Conferences Steering Committee, 2015.