Evaluating Event Credibility on Twitter

Manish Gupta Peixiang Zhao Jiawei Han
University of Illinois at Urbana Champaign

January 7, 2012

Abstract

Though Twitter acts as a realtime news source with people acting as sensors and sending event updates from all over the world, rumors spread via Twitter have been noted to cause considerable damage. Given a set of popular Twitter events along with related users and tweets, we study the problem of automatically assessing the credibility of such events.

We propose a credibility analysis approach enhanced with event graph-based optimization to solve the problem. First we experiment by performing PageRank-like credibility propagation on a multi-typed network consisting of events, tweets, and users. Further, within each iteration, we enhance the basic trust analysis by updating event credibility scores using regularization on a new graph of events.

Our experiments using events extracted from two tweet feed datasets, each with millions of tweets show that our event graph optimization approach outperforms the basic credibility analysis approach. Also, our methods are significantly more accurate (∼86%) than the decision tree classifier approach (∼72%).

1 Introduction

Since its launch in 2006, Twitter has gained huge popularity. 4% of the massive number of tweets constitute news[1]. News tweets are either fresh news (e.g., earthquake reports) or represent social discussions and opinions, related to recent news headlines, which are interesting for event analysis. Opinions expressed by a large group of people on Twitter can be analyzed for effective decision making. Knowledge of hot news events keeps Twitter users up to date with current happenings.

Since the Twitter platform is regarded so important, it becomes necessary to ask, do people consider all content on Twitter as trustworthy? Recent surveys show that just ∼8% people trust Twitter[2], while just ∼12% trust their friends' Twitter streams[3]. Though Twitter has made efforts to enhance trust on their website[4], a lot of rumors[5] have been spread using Twitter in the recent past and have resulted into a lot of damage[6]. A challenging task is to identify such incredible events and prevent them from being promoted, for example, as Twitter Trends. Although tweet feeds contain a lot of signals indicating the credibility, distinguishing non-credible events automatically from the trustworthy ones is clearly a challenging task, primarily due to a lack of any golden truth or instant verification mechanism for recent events.

Incredible events on Twitter could be of two main types: (1) clearly incredible ones (like fake events related to celebrities or strategic locations, partly spiced-up rumors, or erroneous claims made by politicians) and (2) seemingly incredible ones (like ones with informally written tweets, tweets making conflicting claims, tweets lacking any supportive evidence like URLs, tweets without any credibility-conveying words like news, breaking, latest, report, etc.).

Example 1.1. *Common examples of clearly incredible events include "RIP Jackie Chan", "Cables broke on Mumbai's Bandra Worli Sea Link", etc. An event like "24pp" (24 hour panel people show – a comedy show that was broadcasted continuously for 24 hours) can be considered as a seemingly incredible event. This event did not make any news headlines, and different people tweeted about different things they liked/disliked in the show, in a colloquial slang language style.*

In this paper, we study the problem of automatically assessing credibility of Twitter events. Events are like Twitter Trends, which are collections of tweets and can be represented using the Twitter Trend words. Number of followers for a user, presence of URLs in tweets, number of hours for which the event has been

[1] http://www.pearanalytics.com/blog/wp-content/uploads/2010/05/Twitter-Study-August-2009.pdf

[2] http://www.zogby.com/news/ReadNews.cfm?ID=1871

[3] http://www.marketingpilgrim.com/2010/08/trust-the-blog-but-not-the-twitter.html

[4] http://blog.twitter.com/2010/03/trust-and-safety.html

[5] http://news.cnet.com/8301-13577_3-20016906-36.html

[6] http://technology.timesonline.co.uk/tol/news/tech_and_web/the_web/article7069210.ece

discussed are a few important signals related to the attributes of the entities (users, tweets and events) on Twitter, that could be useful for the task. Apart from this, there is a lot of credibility information available in the form of inter-entity credibility relationships. E.g., "with high probability, credible users provide credible tweets" and "similar events should have similar credibility scores". Also, we observe that for a genuine event, tweets are relatively more coherent, compared to non-credible events for which the tweets claim a variety of stories for lack of any authentic evidence or direct experience. The challenge is how to effectively combine all such credibility information effectively? Previous work [4] suggests usage of machine learning classifiers. But (1) they ignore inter-entity relationships completely, and (2) their approach attributes all the features to the *event* entity, while many of the features naturally belong to *tweets* and *users*, rather than to the *events*. To overcome these shortcomings and to exploit the credibility information more effectively, we propose two credibility analysis approaches. Our goal is to assign a credibility score to each event such that more credible events get a higher score.

Our Contributions: (1) To compute the credibility of Twitter events, we propose *BasicCA* which performs PageRank [12]-like iterations for authority propagation on a multi-typed network consisting of events, tweets and users. (2) Next, we propose *EventOptCA* which constructs another graph of events within each iteration and enhances event credibility values using the intuition that "similar events should have similar credibility scores". (3) Using 457 news events extracted from two tweet feed datasets, each with millions of tweets, we show that our methods are significantly more accurate (∼86%) than the classifier-based feature approach (∼72%) for the event credibility assessment task.

Paper Organization: This paper is organized as follows. We provide a formal definition of our problem in Section 2. We discuss the features used by the classifier-based approach in Section 3. Next, we present details of our credibility analysis approach and the event graph optimization in Section 4. We describe our datasets, present our results, and interesting case studies in Section 5. We discuss related work in Section 6. Finally, we summarize our insights and conclude in Section 7.

2 Problem Definition

We study the problem of establishing credibility of events on Twitter. We define an event as follows.

Definition 2.1. (Event) *An event e is specified using a set of required core words R_e and a set of optional subordinate words O_e, $(e = R_e \cup O_e)$. An event can also be considered as a set of tweets.*

For example, consider the event "A bull jumped into the crowds in a Spanish arena injuring 30 people". This event consists of the required words R_e=(bull, crowd) and the optional words O_e=(injured, arena, spain, jumps). Thus the event can be represented as e=(bull, crowd) ∪ (injured, arena, spain, jumps). Here R_e could be Twitter Trend words while O_e can be determined as the set of words occurring frequently in tweets containing all words in R_e.

Credibility is a very broad term and can include different factors for different applications. We provide definitions of credibility of various entities, next. Note that we treat credibility as a score $\in [0, 1]$, rather than a boolean value. For accuracy comparisons, we will use a threshold score to classify events as credible or not.

Credibility of a Twitter event e is the degree to which a human can believe in an event by browsing over a random sample of tweets related to the event. Hence, we can define credibility of an event in terms of its related tweets.

Definition 2.2. (Credibility of an Event $(c_E(e))$) *Credibility of an event e is the expected credibility of the tweets that belong to event e.*

Similarly, we can define the credibility of an user and the tweet as follows.

Definition 2.3. (Credibility of an User $(c_U(u))$) *Credibility of a Twitter user u is the expected credibility of the tweets he provides.*

Definition 2.4. (Credibility of a Tweet $(c_T(t))$) *Credibility of a tweet t is a function of the credibility of related events and users, and the credibility of other tweets that make similar (supporting/opposing) assertions as the ones made by this tweet.*

Problem: Given a set of events along with their related tweets and users, our aim is to find which of the events in the set can be deemed as credible.

Note that our problem setting is quite different from traditional fact finding scenarios. Fig 1 shows a typical fact finder setting where the network consists of three entity types: providers, claims and objects. A provider provides only one claim for an object. Out of multiple and possibly conflicting claims for an object, only one is true and is called the fact. Traditional fact finders [27] aim at computing credibility of sources, and finding most credible claim for each object (Fig 1). Our aim is to find credibility of an event, which is a collection of tweets. Compared to claims, tweets do not make rigorous assertions (though assertions can be extracted from tweets). Also, an object can have only one true claim, while in our setting, an event may consist of many credible tweets (not a particular one). Credible tweets

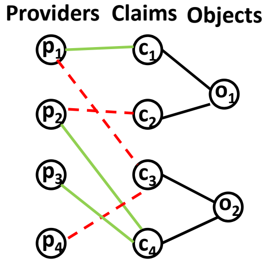


Figure 1: Traditional Providers and Claims Network

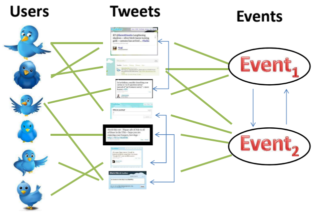


Figure 2: Network Design for BasicCA

could be a rephrasal of each other or may deal with different topics within the event.

3 Classification-Based Approach

In this section, we will introduce a simpler, classifier-based approach to solve the problem of credibility assessment. Besides incorporating all the features that Castillo et al. [4] claimed to be important for the task, we also include a few novel features. With some labeled data, a machine learning classifier model like SVM could be learned using these features.

3.1 User Features Credibility of a user may be influenced by his social reputation and profile completeness, as can be measured using the following factors. (1) Number of friends, followers, and status updates. (2) Is user's Twitter profile linked to its Facebook profile? (3) Is user's account verified by Twitter? (4) When did the user register on Twitter? (5) Does the user profile contain a description, URL, profile image, location?

3.2 Tweet Features Credibility of a tweet may be influenced by the following factors. (1) Is it professionally written tweet with no slang words, '?', '!', smileys, etc.? (2) Does it contain supportive evidence in the form of external URLs? (3) Number of words with first, second, third person pronouns. (4) Is it from the most frequent location related to the event, put up by the user who has put a large number of tweets related to the event, contains the most frequently cited URL related to the event, contains the most frequently used hashtag related to the event? (5) Is it complete i.e., it contains most of the named entities related to the event? A more complete tweet gives a more complete picture of the truth. (6) Does tweet sentiment match with overall sentiment of the event?

3.3 Event Features Credibility of an event may be influenced by the following factors. (1) Number of tweets and retweets related to the event. (2) Number of distinct URLs, domains, hashtags, user mentions, users, locations related to the event. (3) Number of hours for which the event has been popular. (4) Percentage tweets related to the event on the day when the event reached its peak popularity.

3.4 Computation of Features Most of the features mentioned above can be easily computed once we have all the tweets related to an event, and the associated metadata. A tweet is related to an event e, if it contains all words in R_e. Sentiment-based features can be computed using standard sentiment dictionaries (like General Inquirer[7]). We obtained Twitter slang words using online slang dictionaries[8], to compute the "number of slang words" feature.

3.5 What does the Classifier Approach Lack? The classifier approach mentioned in this section fails to work well, mainly because it is not entity and network aware. Features that originally belong to *tweet* and *user* entities are aggregated and attributed to *events*. For example, registration age[9] is a feature of the user, but when using classifier approach, one has to encode it as average registration age of all users who tweet about the event, as has been done in [4]. This is done for all the user and tweet features mentioned above. Thus, the classifier approach, using the above-mentioned features, is not entity-aware. Also, a lot of important information available in the form of inter-entity relationships is not captured by the classifier. This includes intuitive relationships like: (1) With high probability, credible users provide credible tweets; (2) Average credibility of tweets is higher for credible events than that for non-credible events; and (3) Events sharing many common words and topics should have similar credibility scores. In the next section, we will introduce our model and explain how we can deal with these shortcomings.

4 Our Approach

In this section, we will discuss a PageRank-like credibility propagation approach to establish event credibility. We will first present a basic credibility analyzer for Twitter (*BasicCA*) that exploits the basic network information. We enhance the *BasicCA* by performing event graph optimization (*EventOptCA*). We will com-

[7] http://www.wjh.harvard.edu/~inquirer/homecat.htm

[8] http://onlineslangdictionary.com/word-list/,http://www.mltcreative.com/blog/bid/54272/Social-Media-Minute-Big-A-List-of-Twitter-Slang-and-Definitions

[9] A new user has higher probability of being spammy compared to an old user.

pare these algorithms in Section 5.

4.1 Basic Credibility Analysis (BasicCA) The Basic Credibility Analyzer initializes the credibility of different nodes in a network of tweets, users and events (Fig 2) using the results of the classifier developed in Section 3. Using PageRank-like iterations, this initial credibility is propagated all across the network. Every node shares its credibility with its neighbors only, at each iteration. Based on our observations (which we will present in detail in this section), a credible entity links with a higher weight to more credible entities than to non-credible ones. E.g., a credible tweet links to more credible users than to non-credible ones, while a non-credible tweet links to more non-credible users. A credible tweet often links to another credible tweet with a higher positive weight, while it links to non-credible tweets with low (often negative) weights. Hence, every iteration helps in mutually enhancing the credibility of genuine entities and reducing the credibility of non-genuine ones, via propagation. Such credibility propagation hence results in improvement in accuracy over iterations.

Note that though the networks (Fig 1 and 2) look quite similar, our problem setting is quite different, as we explained earlier. Edges in the network are created as follows. User u is linked to tweet t if it puts up that tweet. A tweet t is linked to an event e if it contains all the words in R_e. Edges between tweets and between events denote influence of one tweet (event) on another tweet (event). We will discuss later how we assign weights to these edges.

Propagation of credibility in the network is guided by the following observations which are in line with our definitions of credibility of different entities.

Obs 4.1. *With a high probability, credible users provide credible tweets.*

For example, the tweets from PBS NewsHour[10], BBC-News[11], etc. look reasonably credible, most of the times. Hence, we can express the credibility of a user u as the average of the credibility of the tweets it provides ($N_T(u)$).

$$c_U^i(u) \quad = \quad \frac{\sum_{t\in N_T(u)} c_T^{i-1}(t)}{|N_T(u)|} \tag{4.1}$$

where $N_A(b)$ denotes the neighbors of the entity b of type A.

Obs 4.2. *With a high probability, average credibility of tweets related to a credible event is higher than that of tweets related to a non-credible event.*

Fig 3 shows randomly selected tweets about "Carol Bartz fired". Note that the tweets are written quite professionally. Hence, we can express the credibility of an event e as the average credibility of the tweets it contains ($N_T(e)$).

$$c_E^i(e) \quad = \quad \frac{\sum_{t\in N_T(e)} c_T^{i-1}(t)}{|N_T(e)|} \tag{4.2}$$

Based on these observations and in accordance with our definition, one can infer that credibility of a tweet t can be derived from the credibility of events it discusses ($N_E(t)$) and that of the users who put it up ($N_U(t)$).

$$\begin{aligned} c_T^i(t) \quad &= \quad \frac{\sum_{e\in N_E(t)} c_E^{i-1}(e)}{|N_E(t)|} \\ &+ \quad \rho \times \frac{\sum_{u\in N_U(t)} c_U^{i-1}(u)}{|N_U(t)|} \end{aligned} \tag{4.3}$$

where ρ controls the tradeoff between credibility contributions from related users and events.

4.1.1 Tweet implications A tweet may influence other tweets. We will like to capture these influences when computing credibility of tweets.

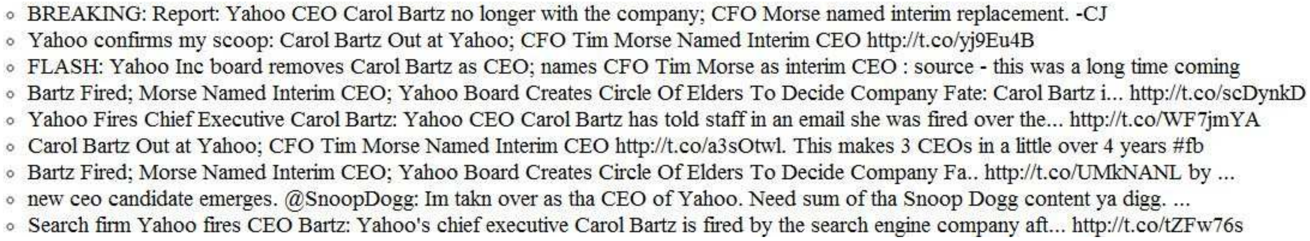
Definition 4.3. (Implication Between Tweets) *Implication value from tweet t' to tweet t, $imp(t' \rightarrow t) \in [-1, 1]$, can be defined as the degree of influence from tweet t' to t. If t' supports t, influence is positive. If t' opposes t, influence is negative.*

$imp(t' \rightarrow t)$ can be computed in two ways: directly or indirectly in terms of claims.

The two step indirect approach: In the first step, mutually exclusive claims could be extracted from tweets using research works that can extract structured records from free text [20, 22]. We define a claim to be an assertion which can be labeled as true or false. In the second step, degree of influence ($infl(t, c) \in [-1, 1]$) of a tweet t towards a claim c can be computed using textual entailment techniques [6, 7]. Next, implication value between a pair of tweets can be computed as the average similarity between the influence offered by the two tweets to different claims. If a tweet neither supports nor opposes the claim, $infl(t, c)$=0. Similarity between t and t' with respect to the claim c would then be $1 - \frac{|infl(t',c) - infl(t,c)|}{2}$. However, such a two step processing has these limitations: (1) Though structured

[10] http://twitter.com/newshour

[11] http://twitter.com/BBCNews



- BREAKING: Report: Yahoo CEO Carol Bartz no longer with the company; CFO Morse named interim replacement. -CJ
- Yahoo confirms my scoop: Carol Bartz Out at Yahoo; CFO Tim Morse Named Interim CEO http://t.co/yj9Eu4B
- FLASH: Yahoo Inc board removes Carol Bartz as CEO; names CFO Tim Morse as interim CEO : source - this was a long time coming
- Bartz Fired; Morse Named Interim CEO; Yahoo Board Creates Circle Of Elders To Decide Company Fate: Carol Bartz i... http://t.co/scDynkD
- Yahoo Fires Chief Executive Carol Bartz: Yahoo CEO Carol Bartz has told staff in an email she was fired over the... http://t.co/WF7jmYA
- Carol Bartz Out at Yahoo; CFO Tim Morse Named Interim CEO http://t.co/a3sOtwl. This makes 3 CEOs in a little over 4 years #fb
- Bartz Fired; Morse Named Interim CEO; Yahoo Board Creates Circle Of Elders To Decide Company Fa.. http://t.co/UMkNANL by ...
- new ceo candidate emerges. @SnoopDogg: Im takn over as tha CEO of Yahoo. Need sum of tha Snoop Dogg content ya digg. ...
- Search firm Yahoo fires CEO Bartz: Yahoo's chief executive Carol Bartz is fired by the search engine company aft... http://t.co/tZFw76s

Figure 3: Average Credibility of Tweets Related to Credible Events is high

claims extraction and textual entailment have been shown to be effective for specific domains, they do not generalize well. (2) They need labeled data, which is expensive to obtain. (3) Deep NLP techniques for the two step processing may be quite time consuming.

The direct approach: Alternatively, one can resort to shallow but efficient unigram features approach. Using such an approach, implication values between two tweets can be computed directly as the TF-IDF or cosine similarity between the two tweets in the space of unigrams. Twitter contains a lot of slang words which can bloat the vocabulary size. Hence, we can compute implications in terms of the most frequent unigrams for the event. This also helps to remove the noise present in tweets in the form of slang words (which form a major part of tweets). Thus, one can compute $imp(t' \rightarrow t)$ for tweets t and t' related to event e in terms of the shared unigrams from the set of event words, as follows.

$$imp(t' \rightarrow t) = \frac{|\{w|w \in t \cap w \in t' \cap w \in e\}|}{|\{w|w \in t' \cap w \in e\}|} \tag{4.4}$$

Recall $e = R_e \cup O_e$.

EXAMPLE 4.4. *Consider the two tweets: t_1=“**Bull jumps** into **crowd** in **Spain**: A **bull jumped** into the stands at a bullring in northern **Spain injuring** at least **30**... http://bbc.in/bLC2qx”. t_2=“Damn! Check out that **bull** that **jumps** the fence into the **crowd**. #awesome”. Event words are shown in bold. Then, implication values can be computed as $imp(t_1 \rightarrow t_2) = 3/6$, $imp(t_2 \rightarrow t_1) = 3/3$.*

Such an approach has been found to be quite effective for news analysis [23] and sentiment analysis [13, 21] tasks. However, sometimes use of negative words like “not” can flip the semantics even though large number of words are shared. The case of such negative words can be handled by making the implication value negative. Also, if the tweet contains words like “hoax”, “rumour”, “fake”, etc., implication value can be negated.

The two step process seems to be more effective than the direct method, but could be time consuming. To be able to detect incredible events on Twitter as quickly as possible, we resort to the direct approach in this work. However, an effective two step approach can only improve the accuracy of our credibility analysis.

After obtaining the tweet implication values, one can rewrite Eq. 4.3 to incorporate the tweet implications and compute tweet credibility as follows.

$$\begin{aligned} c_T^i(t) &= w_{UT} \times \frac{\sum_{u \in N_U(t)} c_U^{i-1}(u)}{|N_U(t)|} \\ &+ w_{TT} \times \sum_{t' \in N_T(t)} c_T^{i-1}(t') \times imp(t' \rightarrow t) \\ &+ w_{ET} \times \frac{\sum_{e \in N_E(t)} c_E^{i-1}(e)}{|N_E(t)|} \end{aligned} \tag{4.5}$$

4.1.2 Computing Weights

In Eq. 4.5, we need to ensure that the credibility contribution from an entity of type A does not overshadow the contribution from another entity of type B. For example, since the number of events is much less than the number of tweets and since we ensure that the credibility vectors always sum up to one, the average credibility value for an event (0.004 for *D2011*) is much higher compared to the average credibility value of a tweet (1.3e-5 for *D2011*). Hence, it is necessary to introduce weights w_{AB}. Weights should be designed to ensure that the credibility contributions are comparable. Thus, the weight should take two factors into account: (1) Number of entities of each type (2) Average number of entities of type A per entity of type B. Thus, $w_{AB} = \frac{|A|}{|B|} \times \frac{1}{avgAPerB}$. For example,

$$w_{UT} = \frac{numUsers}{numTweets} \times \frac{1}{avgUsersPerTweet} \tag{4.6}$$

Weight for the implications term is computed as $w_{AA} = \frac{1}{avgImpA}$. For example,

$$w_{TT} = \frac{1}{avgImpTweets} \tag{4.7}$$

4.1.3 Event Implications

Events are succintly represented using a small set of required and optional words. Degree of similarity between two events can

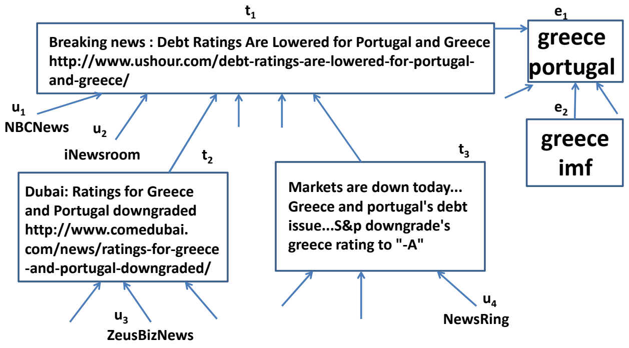

t₁
Breaking news : Debt Ratings Are Lowered for Portugal and Greece http://www.ushour.com/debt-ratings-are-lowered-for-portugal-and-greece/
e₁
greece portugal
e₂
greece imf
u₁
NBCNews
u₂
iNewsroom
t₂
t₃
Dubai: Ratings for Greece and Portugal downgraded http://www.comedubai.com/news/ratings-for-greece-and-portugal-downgraded/
Markets are down today... Greece and portugal's debt issue...S&p downgrade's greece rating to "-A"
u₄
NewsRing
u₃
ZeusBizNews

Figure 4: Credibility Flow Into Event e_1

be estimated using the number of such shared words. Hence, we compute implication value from an event e' to another event e based on the number of the shared required and optional words.

$$imp(e' \to e) = \frac{|(R_{e'} \cup O_{e'}) \cap (R_e \cup O_e)|}{|R_{e'} \cup O_{e'}|} \tag{4.8}$$

Using these event implication values, we can rewrite Eq. 4.2 and compute event credibility as follows.

$$\begin{aligned} c_E^i(e) &= w_{TE} \times \frac{\sum_{t \in N_T(e)} c_T^{i-1}(t)}{|N_T(e)|} \\ &+ w_{EE} \times \sum_{e' \in N_E(e)} c_E^{i-1}(e') \times imp(e' \to e) \end{aligned} \tag{4.9}$$

Fig 4 shows a sub-network focused on event e_1="greece, portugal" (debt ratings lowered). The event e_1 gets credibility from related tweets like t_1 and other events like e_2="(greece, imf)". The tweets in turn derive credibility from other related tweets, related users and events.

4.1.4 Initializing Credibility Vectors Before starting the credibility propagation, how do we initialize the network? The probabilistic results from the classifiers give us a fairly accurate estimate of the credibility of various entities and hence we use them to initialize the credibility of each entity in the network.

$$c_E^0(e) \leftarrow probability_{SVM}(e \text{ is credible}) \tag{4.10}$$

Since, we do not have labeled data for credible users, we cannot learn a classifier using just user features. However, we note down SVM weights SVM_f for all user features (aggregated for every event) $f \in UF$ when learning the classifier to classify events. These SVM weights are an indication of the correlation of various features wrt. the credibility label. Credibility of user u is initialized proportional to the weighted average of normalized user feature values (not aggregated, but per user), $f(u)$, weighted by the SVM weights.

$$c_U^0(u) \leftarrow \sum_{f \in UF} SVM_f \times normalize(f(u)) \tag{4.11}$$

Similarly, we initialize the credibility of tweets as follows.

$$c_T^0(t) \leftarrow \sum_{f \in TF} SVM_f \times normalize(f(t)) \tag{4.12}$$

where TF is the set of tweet features and $f(t)$ is the value of feature f for tweet t.

4.1.5 Discounting Retweets If a user verifies the fact and then retweets, we can consider it to be a full vote for the tweet. Such a retweet can be considered as a form of endorsement. On the other hand, blind retweeting deserves a vote of 0.

OBS 4.5. *a. Often times, users simply retweet without verification because they trust their friends or followees. b. Credibility of a tweet may change when retweeted because of addition or deletion of content.*

EXAMPLE 4.6. *Consider t="Sad news Gazal Maestro Shri Jagjit Singh Ji passed away a Big loss for the music industry R I P ... http://fb.me/EJOdn5p1". It may get retweeted as t′="RT @sukhwindersingh Sad news Gazal Maestro Shri Jagjit Singh Ji passed away a Big loss for the music industry R I P ..". The 140 character limit truncates away the URL, making t′ less credible than t.*

Given a tweet, it is almost impossible to judge if the retweet was done after any verification. Hence, when computing the credibility of any other entity using the credibility of tweets, we multiply the credibility of tweets by $(1 - I(RT) \times RP)$. Here $I(RT)$ is an indicator function which is 1 when the tweet is a retweet and RP is the retweet penalty such that $0 \le RP \le 1$.

4.1.6 Summing Up: BasicCA *BasicCA* as shown in Algorithm 1 initializes the network using classification results (Step 2) and computed implication values (Step 3). Next, it propagates credibility across the entities (Step 5 to Step 10) and terminates when the change in event credibility vector is less than a tolerance value. **Time Complexity:** The number of events are far less compared to number of tweets or users. Also, number of tweets are more than number of users. So, the most expensive step of the algorithm is to compute implications between tweets. Let T be the number of unique tweets, I be the number of iterations, E be the

Algorithm 1 Basic Credibility Analyzer for Twitter (BasicCA)

```
Input: Tweets from multiple users about multiple events.
Init credibility of all entities using classifier results.
Compute implications between tweets and between events.
Compute weights $w_{TT}$, $w_{EE}$, $w_{UT}$, $w_{ET}$ and $w_{TE}$.
while $|c_E^i - c_E^{i-1}|_1 \geq Tolerance$ do
    For (every tweet $t$) Compute $c_T^i(t)$ using Eq. 4.5.
    For (every user $u$) Compute $c_U^i(u)$ using Eq. 4.1.
    For (every event $e$) Compute $c_E^i(e)$ using Eq. 4.9.
    Normalize $c_T^i$, $c_U^i$ and $c_E^i$ such that each vector sums up to one.
end while
return $c_E$: Credibility score for every event $e$.
```

EventType	*D2010*	*D2011*
Non-Credible	260.3	29.4
Credible	374.4	204.6

Table 1: Average Tweet Implication Values

number of events. Then, a tweet can share implication values with T/E tweets (i.e. average number of tweets per event). Thus, the algorithm is $O(IT^2/E)$.

4.1.7 Intuition behind our Credibility Analysis Approach Hoaxes are viral and so a hoax event becomes very popular in a short time, with a large number of users and tweets supporting it. Such signals can confuse the classifier to believe that the event is indeed credible. However, the following factors can help our basic credibility analyzer to identify hoax events as non-credible.

Obs 4.7. *Hoax events are generally not put up by credible users (e.g. A tweet put up by PBS*[12] *will be genuine with a very high probability).*

Thus, credible users do not offer credibility to hoax events, most of the times.

Obs 4.8. *Average implication values between tweets are significantly lower for non-credible events compared to that for credible events.*

Table 1 validates this observation for both of our datasets. For example, for our *D2011* dataset, compared to a high average tweet implications value of 204.6 for credible events, the average tweet implications for non-credible events is merely 29.4. Such an observation is mainly due to the incoherent nature of non-credible events as we explain next. Hoax tweets lack external URLs. People put tweets in their own words, because they have no option to copy paste from some news websites and also lack coherent authentic knowledge. As a result tweets do not make coherent claims. This causes tweet implications to be low, thereby decreasing the credibility of the hoax tweets themselves. Thus, due to lack of credibility contributions from tweets and users, hoax events will tend to have lower credibility scores.

4.2 Performing Event Graph Optimization (EventOptCA)

BasicCA exploits inter-entity relationships and performs better than the classifier approach. However, it considers only weak associations between event credibility scores. Stronger event associations can be inferred based on number of shared unigrams and topics and can be used to compute event similarity. Such event similarity can be exploited with the help of the intuition that similar events should have similar credibility scores. In this section, we will present *EventOptCA* which makes use of this intuition. It performs event credibility updates on a graph of events whose edges use event similarity values as weights. Updates are performed based on regularization of this event graph ensuring that (1) similar events get similar credibility scores, and (2) change in event credibility vector is controlled. We will discuss this approach in detail in this section.

4.2.1 Computing Event Similarity We compute similarity between events e and e' as Jaccard Similarity in terms of shared unigrams and topics as follows.

$$sim(e,e') = \frac{1}{2}\left[\frac{|(R_e \cup O_e) \cap (R_{e'} \cup O_{e'})|}{|(R_e \cup O_e) \cup (R_{e'} \cup O_{e'})|} + \frac{|topics_e \cap topics_{e'}|}{|topics_e \cup topics_{e'}|}\right]$$

Using the two examples below, we discuss how this similarity measure could be effectively used.

Example 4.9. *Consider the events shown in Fig 5. Each of the nodes in this graph corresponds to a Twitter Trend. The weights on the edges denote the similarity between events (in terms of common words and topics). As we can see, weights on edges are relatively high. This is because all of these Twitter Trends are related to the central theme of Cricket World Cup 2011 ('sachin', 'gambhir', etc. are popular players, 'indvspak' was a popular match, 'australia' is a popular cricket-playing nation, etc.) Thus, each of these events should have similar credibility scores.*

One can use this intuition to update the event credibility vector, after every iteration of the credibility analysis. For example, if "gambhir" and "afridi" are marked as credible, but "murali" is marked as incredible, one should be able to exploit the information that "murali" is very similar to "gambhir" and "afridi" and so should be marked as credible.

Example 4.10. *Consider the events "Osama bin Laden dead" and "RIP Steve Jobs". Note that though both the events are about death of a person, the unigrams and topics within each event will be quite different (the first event is related to killing, afghanistan, LeT, pakistan, navy seals while the second event is related to*

[12] http://twitter.com/newshour

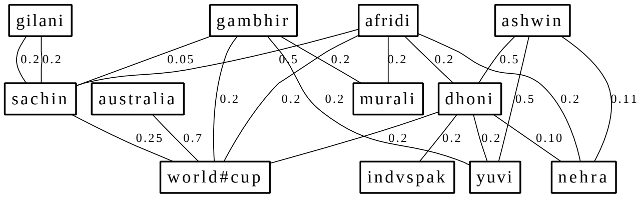


Figure 5: A Subgraph of our *D2011* Event Graph

cancer, iPod, Apple, tech-industry), resulting in lower degree of similarity between the two events.

4.2.2 How to Compute Topics for an Event?

We compute the similarity between events in terms of shared unigrams and topics. Tweets being very short documents, LDA may not be effective for topic detection. Hence, we use the following method for topic detection, inspired by Twitter Monitor [10]. Event $e = R_e \cup O_e$, where R_e is the set of required core words and O_e is the set of optional subordinate words. Let T_e be the set of tweets that contain all words in R_e. For each tweet $t \in T_e$, we find all words present in t and also in O_e. Let S_t be the maximal subset of words from O_e that are also present in t. We compute S_t for every $t \in T_e$. Most frequent S_t's, each unioned with the set R_e, are called topics of the event. Thus, a topic consists of all the required words in the event and a subset of the words from the set of optional words.

Example 4.11. *For the event "oil spill", some topics were: (bp, deepwater, explosion, oil, spill), (bp, oil, spill, marine, life, scientists), (oil, spill, static, kill), (oil, spill, mud, cement, plug) and (9m, gushed, barrels, oil, spill).*

4.2.3 Expressing as a Quadratic Optimization Problem

Given a graph of events, we want to transform the event credibility vector c_E to a new credibility vector c'_E such that (1) similar events get similar credibility scores, and (2) change in event credibility vector is controlled. Let W_{ij} be the weight on edge (i, j).

Then, the above intuitions can be encoded in the form of a quadratic optimization problem (QP) as follows.

$$\begin{aligned} &min\ \frac{1}{2}\sum_{i,j} W_{ij}(c'_E(i) - c'_E(j))^2 + \lambda\frac{1}{2}\sum_i (c'_E(i) - c_E(i))^2 \\ &s.t.\ -1 \le c'_E(i) \le 1\ \ \forall i \in \{1, ..., n\} \end{aligned} \tag{4.13}$$

The first term in the objective function in Eq. 4.13 ensures that similar events should have similar credibility scores. When W_{ij} is high, i.e., the two events are very similar, the minimization tries to assign similar credibility scores to the two events. The second term ensures that the change in event credibility vector should be as less as possible. λ controls the tradeoff between smoothing applied by the first term and the amount of change in the credibility vector.

For the QP to be feasible, we first shift the event credibility vector values from [0,1] to [-1,1] and also expect the new credibility vector to have values in the interval [-1,1]. After the optimization, we shift the c'_E values back from [-1,1] to [0,1] so that they can be used for further iterations of the credibility analysis.

Eq. 4.13 can be rewritten in matrix form as follows.

$$\begin{aligned} &min\ \frac{1}{2}c'^T_E(2(D - W) + \lambda I)c'_E + (-\lambda c^T_E.c'_E) \\ &s.t.\ -1 \le c'_E(i) \le 1\ \ \forall i \in \{1, ..., n\} \end{aligned} \tag{4.14}$$

where D is the diagonal matrix such that $D_{ii} = \sum_j W_{ij}$. This can be further simplified as follows.

This QP can be easily solved using solvers like quadprog in Matlab [11].

Lemma 4.12. *The QP problem (Eq. 4.14) has a unique global minimizer.*

Proof. We omit the rigorous proof for lack of space. However the proof can be easily reconstructed using the following discussion. The terms representing the matrix $2(D-W)+\lambda I$ are square terms (compare Eq. 4.13 and 4.14), and hence will always be positive. Hence, the matrix $2(D - W) + \lambda I$ is positive definite. This ensures that the QP is feasible and will converge to a unique global minima.

4.2.4 Summing Up: EventOptCA

EventOptCA as shown in Algorithm 2 initializes the network using classification results. Next, it propagates credibility across the entities and terminates when the change in the event credibility vector is less than a threshold value. Within each such iteration, it creates a separate graph of events and performs update to the event credibility vector using the quadratic optimization.

Time Complexity: The algorithm is $O(IT^2/E+E^n)$; solving the QP is polynomial in the number of variables (events), where n is typically a small constant. However, since the number of events is generally very small, solving the QP is relatively faster than tweet credibility computations.

5 Experiments

In this section, we will first describe our datasets. Then we will present our results, which demonstrate the effectiveness of our methods. We will conclude with interesting case studies.

Algorithm 2 Event Graph Optimization-Based Credibility Analyzer (EventOptCA)

```
Input: Tweets from multiple users about multiple events.
Init credibility of all entities using classifier results.
Compute event similarity matrix $W$.
Compute implications between tweets and between events.
Compute weights $w_{TT}$, $w_{EE}$, $w_{UT}$, $w_{ET}$ and $w_{TE}$.
while $|c_E^i - c_E^{i-1}|_1 \geq Tolerance$ do
    For (every tweet $t$) Compute $c_T^i(t)$ using Eq. 4.5.
    For (every user $u$) Compute $c_U^i(u)$ using Eq. 4.1.
    For (every event $e$) Compute $c_E^i(e)$ using Eq. 4.9.
    Normalize $c_T^i$, $c_U^i$ and $c_E^i$ such that each vector sums up to one.
    Construct a graph of events.
    Update $c_E^i$ using the quadratic optimization mentioned in Eq. 4.13.
    Normalize $c_E^i$ such that each vector sums up to one.
end while
return $c_E$: Credibility score for every event $e$.
```

5.1 Datasets We use two datasets: *D2010* and *D2011*. Events for *D2010* were supplied by Castillo [4]. They extracted events using Twitter Monitor [10] from tweet feeds for Apr-Sep 2010. The original dataset had 288 news events, along with R_e and O_e for each e. After removing events with less than 10 tweets, we were finally left with 207 events (of which 140 are labeled as credible).

For *D2011* dataset, we obtained tweet feeds for Mar 2011 using the Twitter Feeds API[13]. We obtained the core required words for events from the Twitter Trends API [14]. We removed events with less than 100 tweets. We labeled these events as social gossip or news, and then use 250 of the news events (of which 167 are labeled as credible) for our study. Subordinate words (O_e) for the events are obtained by finding the most frequent words in tweets related to the event (which do not belong to the core required word set). To have a prominent network effect and also to reduce the number of tweets per event, we sampled tweets such that there are many tweets from relatively small number of users.

Table 2 shows the details for our datasets. For both the datasets, we used the standard General Inquirer[15] sentiment dictionaries for sentiment analysis of tweets. We obtained Twitter slang words using online slang dictionaries[16]. Topics were obtained for the events using the set of subordinate words as mentioned in Section 4.

When computing credibility of different entities, we use different weights (shown in Table 3) such that the credibility contributions are comparable.

Statistic	*D2010*	*D2011*
#Users	47171	9245
#Tweets	76730	76216
#Events	207	250
#PosEvents	140	167
#NegEvents	67	83
#Topics	2874	2101

Table 2: Dataset Details

	D2010	*D2011*
w_{TT}	0.003	0.007
w_{UT}	0.431	0.117
w_{ET}	0.003	0.003
w_{EE}	2.902	4.823
w_{TE}	1.503	1.390

Table 3: Weights for Credibility Analysis

	Classifier		BasicCA		EventOptCA	
	D2010	D2011	D2010	D2011	D2010	D2011
Accuracy	72.46	72.4	76.9	75.6	**85.5**	**86.8**
False +ves	36	41	24	23	28	31
False −ves	21	28	22	28	2	2

Table 4: Accuracy Comparison

Labeling: For labeling events in *D2011*, we obtained news headlines by scrapping RSS feeds for different news categories from top ten news websites including Google News, Yahoo! News, PBS, BBC, etc. For *D2010*, we used the Google News Archives[17] while labeling. In the first step of labeling, we removed non-English and social gossip events. Next, we identified if the event was present in news headlines for the day when the event reached popularity. An event is searched in the news headlines using an AND/OR query. All the core required words should be present while the subordinate optional words may or may not be present. Then we judged the credibility of event based on randomly selected ten tweets for the event. Events were thus labeled as "hoax", "not in news", "might be false", or "certainly true", in the order of increasing credibility. In case an event can belong to multiple classes, we labeled it to belong to the less credible class. For example, an event may not be in news and can be a hoax, in that case, we labeled it as hoax. Finally, we consider "certainly true" events as credible, the remaining categories as non-credible.

5.2 Accuracy Comparison We show the accuracy results for the two datasets using different approaches in Table 4. For the classifier-based approach we tried various classifiers: SMO (SVM), Naïve Bayes, KNN (IBk), and decision trees (J48). Decision trees perform the best for *D2010*, while KNN works best for *D2011*. On an average, we observed that the classifier-based approach provides just ∼72% accuracy. Our basic credibility analyzer approach provides 3-4% boost in accuracy. Updating the event vector using the event graph-based optimization provides ∼14% boost in accuracy over the baseline. *EventOptCA* does result into a few more false positives compared to the basic credibility analyzer, but overall it reduces a lot of false negatives. For these results, we used λ=1. We varied RP from 0 to 1 and found that the results remained almost the same for RP >0.3. For our experiments, we use RP=0.75.

Varying λ: We varied λ to study sensitivity of the accuracy to the parameter. We show the results in Table 5. Although the accuracy is quite insensitive to

[13] https://stream.twitter.com/1/statuses/sample.json

[14] http://api.twitter.com/1/trends/daily.json

[15] http://www.wjh.harvard.edu/~inquirer/homecat.htm

[16] http://onlineslangdictionary.com/word-list/,http://www.mltcreative.com/blog/bid/54272/Social-Media-Minute-Big-A-List-of-Twitter-Slang-and-Definitions

[17] http://news.google.com/news/advanced_news_search

Parameter λ	*D2010*	*D2011*
0.25	83.1	82.4
0.5	84.4	86
0.75	86.5	84.4
1	85.5	86.8
5	81.6	86
20	78.7	84.8
50	78.3	82.8

Table 5: Varying the Parameter λ

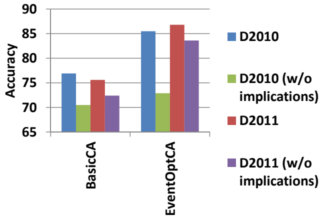


Figure 6: Effect of Removing Implications

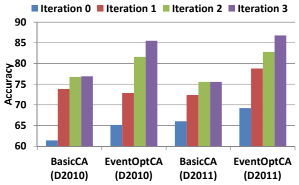


Figure 7: Accuracy Variation (with respect to #Iterations)

D2010	*D2011*
NewsCluster	funkmasterflex
NewsRing	BreakingNews
oieaomar	cwcscores
portland_apts	RealTonyRocha
newsnetworks	espncricinfo
Newsbox101	GMANewsOnline
euronewspure	ipadmgfindcom
nevisupdates	aolnews
_Reality_Check_	QuakeTsunami
cnnbrk	TechZader

Table 8: Most Credible Users

the parameter values from 0.25 to 5, the best accuracy is achieved at values between 0.5 and 1.

5.3 Important Features We use the SVM weights for computing the initial credibility values for different entities in the trust network. Tables 6 and 7 show the top five features for tweets, users and events. Note that though some features appear consistently at the top for both the datasets, many of them do not. This indicates that the classifier-based approach is not stable across datasets.

5.4 Effect of Removing Implications While performing credibility analysis, we consider implications between entities of the same type. To measure the impact of implications on the accuracy achieved, we performed experiments by removing implication values. We show the results for both of our credibility analyzer approaches in Fig 6. The results show that implications are important. Without considering implications, we notice a significant loss in accuracy.

5.5 Accuracy Variation (with respect to #Iterations) As shown in Fig 7, we observe that the accuracy improves per iteration for 3-4 iterations. After that the accuracy stabilizes for *BasicCA*. This is because our *BasicCA* use PageRank-like iterations, and hence displays Markovian behaviour. However, besides PageRank-like authority propagation, *EventOptCA* also performs an optimization step. Hence, for *EventOptCA*, convergence cannot be guaranteed. However, for 3-4 iterations, we observe that the accuracy improves per iteration for both of our datasets.

5.6 Case Studies Table 8 shows the top most credible users for the two datasets. Note that a lot of the users at the top are news agencies. These are the users with a lot of followers and status updates. A few times, we also find other influential users like celebrities (like funkmasterflex) which have a lot of followers or automated tweeting users (like portland_apts) which put in a lot of tweets.

Next, we present a few examples which demonstrate the effectiveness of our approaches.

EventOptCA exploits similarity between events. The event "wembley" is quite similar to the events "englandvsghana" and "ghana". There was a friendly football match between Ghana and England at the Wembley Stadium. Though *BasicCA* incorrectly marked "wembley" as non-credible, *EventOptCA* raised the credibility of "wembley" based on the high credibility of "englandvsghana" and "ghana", and hence marked it as credible.

Similarly, based on high credibility values for "alonso" and "barrichello", the event "bbcf1" was correctly marked as credible. Alonso and Barrichello were racing drivers who participated at the Hungarian Grand Prix (BBC F1 race).

There were also examples of events marked non-credible by the classifier like "alonso" (The racing driver was running 3rd in Hungarian Grand Prix.) but marked correctly by *BasicCA*.

"24pp" is a Twitter Trend event. The classifier marked it as a credible event. However, the tweets for the event as shown below do not look quite credible.

- #24pp had enough - I can take small doses of all the people on the show but not that much.
- #24pp is the best way to spend a saturday :)
- #24pp Sir David Frost is hilarious!!
- Andy Parsons is being fucking brilliant on this #24PP edition of #MockTheWeek . . . shame no-one else is really . . .
- back link habits http://emap.ws/7/9Chvm4 DMZ #24pp Russell Tovey

The words in $R_e \cup O_e$ for the "24pp" event were (24pp, rednoseday, bbccomedy, jedward, panel, tennant, trending, david, walliams, russell). Note that these words do not occur very frequently across tweets for the

User Features	Tweet Features	Event Features
UserHasProfileImage?	TweetWithMostFrequentUserMentioned?	CountDictinctDomains
UserRegistrationAge	LengthInChars	CountDistinctUsers
UserFollowerCount	TweetWithMostFreqURL?	CountDistinctHashtags
UserStatusCount	TweetFromMostFreqLoc?	NumberOfUserLocations
UserHasLocation?	URLfromTop10Domains?	NumberOfHours

Table 6: Important Features for *D2010*

User Features	Tweet Features	Event Features
VerifiedUser?	NumURLs	CountDictinctDomains
UserStatusCount	TweetSentiment	CountDictinctURLs
UserHasFacebookLink?	TweetsWithMostFrequentUserMentioned?	CountDistinctHashtags
UserHasDesc?	ExclamationMark?	PercentTweetsOnPeakDay
UserFollowerCount	NumHashTags?	CountDistinctUsers

Table 7: Important Features for *D2011*

"incoherent" event. This causes low tweet implication values. As a result the average credibility of these tweets is very low and the event gets correctly marked as non-credible by *BasicCA*.

5.7 Performance Considerations Millions of tweets are posted on Twitter everyday. Majority of these tweets are social gossip (non-newsworthy). Our system assumes an input of newsworthy events, e.g., from Twitter Trends. On a particular day, there could be up to 1000 Twitter Trends. Running our algorithm on our datasets of 200–250 events takes a few minutes. Thus, processing all the Twitter Trends every day would not take more than a few minutes. In other words, our algorithm can be used practically in real-time systems.

5.8 Drawbacks of our Methods Our methods can be improved further based on these observations. (1) Deep NLP techniques can be used to obtain more accurate tweet implications. (2) Evidence about an event being rumorous may be present in tweets themselves. However, deep semantic parsing of tweets may be quite inefficient. (3) Entertainment events tend to look not so credible, as the news is often written in a colloquial way. Predicting credibility for entertainment events may need to be studied separately.

6 Related Work

Our paper is related to work in the areas of credibility analysis of online social content, credibility analysis for Twitter, fact finding, and graph regularization.

Credibility Analysis of Online Social Content: The perceived credibility of web content can depend on a lot of features comprising those of the reader (Internet-savvy nature, politically-interested, experts, etc.), the author (gender, author's image), the content (blog, tweet, news website), and the way of presenting the content (e.g. presence of ads, visual design). 13% of the articles were found to contain mistakes [5] on Wikipedia. Online news content has been found to be less credible than newspaper and television news [1]. Blogs are considered less trustworthy than traditional news websites [24] except for politics and tourism.

Credibility Analysis for Twitter: Twitter has been used for tracking epidemics, detecting news events, geolocating such events [17], finding emerging controversial topics [15], locating wildfires, hurricanes, floods, earthquakes, etc. However, users often do not believe in tweets [18] because (1) often people have not even met those mentioned as their friends on Twitter, (2) tracing original user who tweeted about something is difficult, and (3) changing information along the tweet propagation path is very easy.

Warranting is a method where users decide whether other users are trustworthy based on what their friends say about them. Schrock [19] observed that "warranting" does not work on Twitter. Truthy[18] service from researchers at Indiana University, collects, analyzes and visualizes the spread of tweets belonging to "trending topics" [16] using crowd-sourcing.

Automated information credibility detection on Twitter has been recently studied in [4] using a supervised classifier-based approach. They develop two classifiers: the first one classifies an event as news versus chat, while the second one classifies a news event as credible or not. We presented a first work on using principled credibility analysis approach for the problem of establishing credibility for microblogging platforms. Unlike [4], where the features can be assigned only for events, our approach is entity-type-aware and exploits inter-entity relationships.

Fact Finding Approaches: Recently there has been a lot of work in the data mining community on performing trust analysis based on the data provided by multiple sources for different objects. Yin et al. [27] introduced a fact finder algorithm *TruthFinder* which performs trust analysis on a providers-claims network. This work was followed by some more fact finder algorithms: *Sums, Average.Log, Investment, Pooled Investment* by Pasternack et al. [14]. A large body of

[18] http://truthy.indiana.edu/

work [2, 3, 8, 23, 26] has been done further, in this area.

Graph Regularization: Graph regularization is a method for performing smoothing over a network. It has been mainly studied in semi-supervised settings for applications like web-page categorization [28], large-scale semi-definite programming [25], color image processing [9], etc. We used a similar approach for our problem, to smooth out the event credibility values.

7 Conclusion

In this work, we explored the possibility of detecting credible events from Twitter feeds using credibility analysis. We used a new credibility analysis model for computing credibility of linked set of multi-typed entities. We exploited (1) tweet feed content-based classifier results; (2) traditional credibility propagation using a simple network of tweets, users and events; and (3) event graph-based optimization to assign similar scores to similar events. Using two real datasets, we showed that credibility analysis approach with event graph optimization works better than the basic credibility analysis approach or the classifier approach.

8 Acknowledgments

Thanks to Twitter for making the tweet feeds available, to Carlos Castillo for sharing their event dataset, and to anonymous reviewers for their insightful comments. Research was sponsored by Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-09-2-0053 and NSF IIS-09-05215. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

References

[1] R. Abdulla, B. Garrison, M. Salwen, and P. Driscoll. The Credibility of Newspapers, Television News, and Online News. *Association for Education in Journalism and Mass Communication*, Jan 2002.

[2] R. Balakrishnan. Source Rank: Relevance and Trust Assessment for Deep Web Sources based on Inter-Source Agreement. In *Proc. of the Intl. Conf. on World Wide Web (WWW)*, pages 227–236. ACM, 2011.

[3] L. Berti-Equille, A. D. Sarma, X. Dong, A. Marian, and D. Srivastava. Sailing the Information Ocean with Awareness of Currents: Discovery and Application of Source Dependence. In *Proc. of the Conf. on Innovative Data Systems Research (CIDR)*. www.crdrdb.org, 2009.

[4] C. Castillo, M. Mendoza, and B. Poblete. Information Credibility on Twitter. In *Proc. of the Intl. Conf. on World Wide Web (WWW)*, pages 675–684. ACM, 2011.

[5] T. Chesney. An Empirical Examination of Wikipedia's Credibility. *First Monday*, 11(11), 2006.

[6] I. Dagan and O. Glickman. Probabilistic Textual Entailment: Generic Applied Modeling of Language Variability. In *Proc. of the Learning Methods for Text Understanding and Mining*, Jan 2004.

[7] I. Dagan, O. Glickman, and B. Magnini. The PASCAL Recognising Textual Entailment Challenge. In *Machine Learning Challenges*, volume 3944, pages 177–190. Springer, 2006.

[8] M. Gupta, Y. Sun, and J. Han. Trust Analysis with Clustering. In *Proc. of the Intl. Conf. on World Wide Web (WWW)*, pages 53–54, New York, NY, USA, 2011. ACM.

[9] O. Lezoray, A. Elmoataz, and S. Bougleux. Graph Regularization for Color Image Processing. *Computer Vision and Image Understanding*, 107(1–2):38–55, 2007.

[10] M. Mathioudakis and N. Koudas. TwitterMonitor : Trend Detection over the Twitter Stream. *Proc. of the Intl. Conf. on Management of data (SIGMOD)*, pages 1155–1157, 2010.

[11] MATLAB. *Version 7.9.0.529 (R2009b)*. The MathWorks Inc., Natick, Massachusetts, 2009.

[12] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. In *Proc. of the Intl. Conf. on World Wide Web (WWW)*, pages 161–172. ACM, 1998.

[13] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: Sentiment Classification using Machine Learning Techniques. In *Proc. of the Empirical Methods in Natural Language Processing (EMNLP)*, pages 79–86. ACL, 2002.

[14] J. Pasternack and D. Roth. Knowing What to Believe (When You Already Know Something). In *Proc. of the Intl. Conf. on Computational Linguistics (COLING)*. Tsinghua University Press, 2010.

[15] A. M. Popescu and M. Pennacchiotti. Detecting Controversial Events From Twitter. In *Proc. of the ACM Intl. Conf. on Information and Knowledge Management (CIKM)*, pages 1873–1876. ACM, 2010.

[16] J. Ratkiewicz, M. Conover, M. Meiss, B. G. calves, S. Patil, A. Flammini, and F. Menczer. Detecting and Tracking the Spread of Astroturf Memes in Microblog Streams. *arXiv*, Nov 2010.

[17] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake Shakes Twitter users: Real-Time Event Detection by Social Sensors. In *Proc. of the Intl. Conf. on World Wide Web (WWW)*, pages 851–860. ACM, 2010.

[18] M. Schmierbach and A. Oeldorf-Hirsch. A Little Bird Told Me, so I didn't Believe It: Twitter, credibility, and Issue Perceptions. In *Proc. of the Association for Education in Journalism and Mass Communication*. AEJMC, Aug 2010.

[19] A. Schrock. Are You What You Tweet? Warranting Trustworthiness on Twitter. In *Proc. of the Association for Education in Journalism and Mass Communication*. AEJMC, Aug 2010.

[20] S. Soderland. Learning Information Extraction Rules for Semi-Structured and Free Text. *Machine Learning*, 34:233–272, 1999.

[21] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. In *Proc. of the Intl. AAAI Conf. on Weblogs and Social Media (ICWSM)*. The AAAI Press, 2010.

[22] P. Viola and M. Narasimhan. Learning to Extract Information from Semi-Structured Text using a Discriminative Context Free Grammar. In *Proc. of the Intl. ACM Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 330–337. ACM, 2005.

[23] V. V. Vydiswaran, C. Zhai, and D. Roth. Content-Driven Trust Propagation Framework. In *Proc. of the Intl. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pages 974–982. ACM, 2011.

[24] C. R. WebWatch. Leap of Faith: Using the Internet Despite the Dangers, Oct 2005.

[25] K. Q. Weinberger, F. Sha, Q. Zhu, and L. K. Saul. Graph Laplacian Regularization for Large-Scale Semidefinite Programming. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2007.

[26] X. Yin and W. Tan. Semi-Supervised Truth Discovery. In *Proc. of the Intl. Conf. on World Wide Web (WWW)*, pages 217–226. ACM, 2011.

[27] X. Yin, P. S. Yu, and J. Han. Truth Discovery with Multiple Conflicting Information Providers on the Web. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 20(6):796–808, 2008.

[28] T. Zhang, A. Popescul, and B. Dom. Linear Prediction Models with Graph Regularization for Web-page Categorization. In *Proc. of the Intl. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pages 821–826. ACM, 2006.