RAProp: Ranking Tweets by Exploiting the Tweet/User/Web Ecosystem and Inter-Tweet Agreement

Srijith Ravikumar[†], Kartik Talamadupula[†], Raju Balakrishnan[§], Subbarao Kambhampati[†]

[†]Dept. of Computer Science and Engg.
Arizona State University
Tempe AZ 85287
{srijith,krt,rao} @ asu.edu

[§]Groupon, Inc.
3101 Park Blvd
Palo Alto CA 94306
raju @ groupon.com

ABSTRACT

The increasing popularity of Twitter renders improved trustworthiness and relevance assessment of tweets much more important for search. However, given the limitations on the size of tweets, it is hard to extract measures for ranking from the tweets' content alone. We present a novel ranking method called *RAProp*, which combines two orthogonal measures of relevance and trustworthiness of a tweet. The first, called Feature Score, measures the trustworthiness of the *source* of the tweet by extracting features from a 3-layer Twitter ecosystem consisting of users, tweets and webpages. The second measure, called agreement analysis, estimates the trustworthiness of the *content* of a tweet by analyzing whether the content is independently corroborated by other tweets. We view the candidate result set of tweets as the vertices of a graph, with the edges measuring the estimated agreement between each pair of tweets. The feature score is propagated over this agreement graph to compute the top-k tweets that have both trustworthy sources and independent corroboration. The evaluation of our method on 16 million tweets from the TREC 2011 Microblog Dataset shows that for top-30 precision, we achieve 53% better precision than the current best performing method on the data set, and an improvement of 300% over current Twitter Search.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval models, Selection process*

Keywords

Twitter, trust, relevance, agreement, search, microblog

1. INTRODUCTION

Twitter, the popular microblogging service, is increasingly being looked upon as a source of the latest news and trends. The open nature of the platform, as well as the lack of restrictions on who can post information on it, leads to fast dissemination of all kinds of information on events ranging from breaking news to very niche occurrences. However, the popularity of Twitter has led to increased incentives for abusing and circumventing the system, and this is manifested as microblog spamming. The open nature of Twitter proves to be a double-edged sword in such scenarios, and leaves it extremely vulnerable to the propagation of false information from profit-seeking and malicious users (*cf.* [19, 24, 25]).

Unfortunately, Twitter's native search does not seem to consider the possibility of users crafting malicious tweets, and instead only considers the presence of query keywords in, and the temporal proximity (recency) of, tweets [26]. The current Twitter search considers the recency of the tweet to be the single most important metric for judging relevance. Though we believe the recency of a tweet *may* be an indicator of relevance (a tweet in the last couple of hours may be more relevant than a tweet a week ago), it may not be the sole relevance metric for ranking. For example, for a query "`White House spokesman replaced`" the top-5 tweets returned by Twitter Search are as shown in Figure 3a. The tweets are the most recent tweets at query time, and contain one or more of the query terms; notice that none of these five results seem to be particularly relevant to the query. Straightforward improvements such as adapting TF-IDF ranking to Twitter unfortunately do not improve the ranking. On Twitter, it is common to find tweets that contain just the query terms, with no other useful context or information. TF-IDF similarity fails to penalize these tweets. A closer inspection shows that the only relevant tweet (5^{th} tweet) is from a credible news source which points to a web page that is also trustworthy. Thus, the user/web features of a tweet may be considered just as important as the query similarity in order to determine the relevance to a query.

1.1 Our Method: RAProp

We believe that to improve the ranking of tweets, we must take into account the trustworthiness of tweets as well. Our method – *RAPRop* – combines two orthogonal measures of relevance and trustworthiness of a tweet. The first, called the Feature Score, measures the trustworthiness of the *source* of the tweet. This is done by extracting features from a 3-layer Twitter ecosystem, consisting of users, tweets and the pages referred to in the tweets. The second measure, called agreement analysis, estimates the trustworthiness of the *content* of a tweet, by analyzing whether the content is independently corroborated by other tweets.

CIKM'13, October 27 - November 01 2013, San Francisco, CA, USA
Copyright 2013 ACM 978-1-4503-2263-8/13/10 ...$15.00.
http://dx.doi.org/10.1145/2505515.2505667 .

We view the candidate result set of tweets as the vertices of a graph, with the edges measuring the estimated agreement between each pair of tweets. The feature score is propagated over this agreement graph to compute the top-k tweets that have both trustworthy sources and independent corroboration.

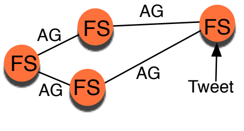


Figure 1: *Propagation of Feature Sores(FS) over Agreement Graph (AG).*

In the next section, we explain how we use the user, web and tweet features to formulate a Feature Score for each tweet. We explain in Section 3 how we measure the popularity of a topic using pairwise Agreement. Then, in Section 4, we explain how we rank our tweets using the Feature Score and agreement graph generated via the methods in the preceding sections. Section 5 presents our evaluation. We conclude with an overview of related work.

2. FEATURE SCORE

In order to compute the trustworthiness of the source of a tweet, we model the entire Twitter ecosystem as a three layer graph as shown in Figure 2. Each layer in this model corresponds to one of the characteristics of a tweet – the content, the user, and the links that are part of that tweet. The user layer consists of the set U of all users u such that a tweet t_u by the user u is returned as part of the candidate result set R for the query. Since the user base of twitter is growing exponentially, we believe that our user trustworthiness algorithm needs a good predictor of the trustworthiness of unseen users profiles. Hence, instead of computing user trustworthiness score from the follower-followee graph [27], we compute the trustworthiness of a user from the user profile information. The user features that we use are: *follower count, friends count, whether that user (profile) is verified, the time since the profile was created, and the total number of statuses (tweets) posted by that user.* Another advantage of computing trustworthiness of a user from the user profile features is that we can adjust our trustworthiness score in accordance with changes to the profile (e.g.. an increase in the number of followers) more quickly.

The tweet layer consists of the content of the tweets in R. We select some features of a tweet that are found to do well in determining the trustworthiness of that tweet [7]. The features we pick include: *whether the tweet is a re-tweet; the number of hash-tags; the length of the tweet; whether the tweet mentions a user; the number of favorites received; the number of re-tweets received; and whether the tweet contains a question mark, exclamation mark, smile or frown.* To these features, we add a feature of our own: TF-IDF similarity which is weighted by proximity of the query keywords in the tweet. Our intuition is that a tweet that contains most of the query terms may be more relevant to the query than a tweet that contains only one of the query terms. Proximity of the query keywords in the tweet is a very important feature when judging the relevance due to the low likelihood of repeating of query terms in the tweet. We try to account for this in our TF-IDF similarity score by exponentially decaying the TF-IDF similarity based on the proximity of the query terms in the tweet as the following: $S = \mathrm{T(t_i, Q)} \times e^{\frac{-w \times d}{l}}$, where $T(t_i, Q)$ is the TF-IDF similarity of the tweet t_i to the query Q; $w = 0.2$ is a constant (empirically decided on a sample data set) that sets the weight for the proximity score; l is the number of terms in the query; and d is the sum of distances between each term in the query to its nearest neighbor.

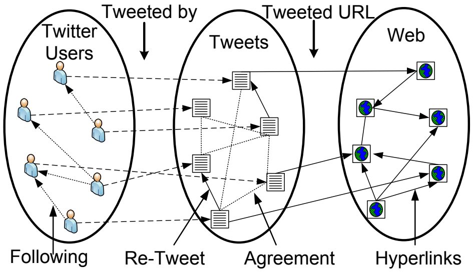


Figure 2: *Three layer ecosystem of Twitter space composed of user layer, tweets layer and the web layer*

The web/link layer consists of the links that are used in tweets. The web has an existing, easily queryable repository that scores web pages based on some notion of trust and influence – PageRank. For each tweet that contains a web link, we instantiate a node that represents that link in the web layer of the graph. There are links from that tweet to the node in the web layer, as well as intra-layer links among the nodes in the web layer based on link relationships on the open web.

The proposed ranking is performed in the tweets layer, but all three layers are used to compute what we call the *Feature Score*. The features from the user and the web page are linked to the tweets by the "Tweeted by" relation and "Tweeted URL" relation.

2.1 Computing Feature Score

To learn the Feature Score from features, we use a Random Forest based learning to rank method [5]. Random Forest is an ensemble learning based classifier that creates multiple decision forests at training time using the bagging approach. We train the Random Forest with the User, Tweet and Web features described previously. We used the gold standard relevance values (described in Section 5.2) for training and testing our model. 5% of the gold standard dataset was randomly picked for training the model, and another 5% to test the trained model (the remaining data is reserved for the experiments). Since we do not want to penalize tweets that do not contain a URL, or user information that we were unable to crawl, we impute the missing feature values using population average. We normalize the Feature Score to lie between 0 and 1.

Here we look back again at our example query's ("`White House spokesman replaced`") results ranked using just Feature Score in Figure 3b. We notice that in the top-5 results, only one tweet is relevant to the query, and the rest of the tweets are about other topics that just contain part of the query terms. In the following section, we look into finding tweets whose content is replicated by a large pool of independent, trustworthy users.

3. AGREEMENT

The Feature Score is more of a measure of the trustworthiness of the user/web page and popularity of the tweet, rather than the trustworthiness of the content itself. Just as the popularity of a tweet is measured by the number of re-tweets it gets, the popularity of the tweet's content may be measured by the number of independent trustworthy users who endorse that content. Although the re-tweet relations among Twitter messages can be seen as endorsement, they fall far short both because of their sparsity and because they do not capture topic popularity. In this section, we develop a complementary endorsement structure among tweets by interpreting mutual agreement between two tweets as an implicit endorsement.

3.1 Agreement as a Metric for Popularity & Trust

Given the scale of Twitter, it is quite normal for the set of tweets returned by a query to contain tweets about multiple topics. The user is likely to be interested in only a few of these topics. Due to the temporal nature of Twitter [23], we hypothesize that the most popular topic is more likely to be about breaking news. Hence, tweets from a popular topic are more likely to be relevant to the user. We use the pair-wise agreement as votes in order to measure the topic popularity. Using agreement as a metric to measure popularity of a topic may be seen as a logical extension of using re-tweets to measure the popularity of a tweet. This method has been found to perform well [4] on the deep web; if two independent users agree on the same fact – that is, they tweet the same thing – it is likely that the content of those tweets is trustworthy. As the number of users who tweet semantically similar tweets increases, so does the belief in the idea that those tweets are all trustworthy.

3.2 Agreement Computation

Computing the pair-wise semantic agreement (as outlined above) between tweets at query-time, while still satisfying timing and efficiency concerns, is a challenging task. Due to this, only computationally simple methods can be realistically used. TF-IDF similarity has been found to perform well when measuring semantic similarity for named entity matching [9] and for computing semantic similarity between web database entities [4]. In web scenarios, the IDF makes sure that more common words such as verbs are weighted lower than nouns which are less frequent. However, due to the sparsity of verbs and other stop words in tweets, we notice that the IDF for some verbs tends to be much higher than nouns and adverbs. Hence, we weight the TF-IDF similarity for each part of speech differently – the intent is to weigh the tags that are important to agreement higher than other tags. We use a Twitter POS tagger [10] to identify the parts of speech in each tweet. The agreement of a pair of tweet T_1,T_2 is defined as $AG(T_1, T_2) = \sum_{t \in (T_1 \cap T_2)} TF(t_1) \times TF(t_2) \times IDF(t)^2 \times P(t)$, where $P(t)$ is set by us (empirically by testing on a sample data set) such that we give higher weights to POS that determine that the tweets are about the same topic as the URL (8.0), Hash tags(6.0), Proper noun(4.0), Common noun / Adjective / Adverb (3.0) and lesser weights to other POS that are less indicative of the agreement between the tweets such as Numerical (2.0), Pronoun / Verb (1.0), Interjection / Preposition (0.5), and Existential(0.2).

We compute TF-IDF similarity on the stop word removed and stemmed candidate set, R_Q. However, due to the way Twitter's native search (and hence our method, which tries to improve it) is set up, every single result $r \in R_Q$ contains one or more of the query terms in Q. Thus the actual content that is used for the agreement computation – and thus ranking – is actually the *residual content* of a tweet. The residual content is that part of a tweet which does not contain the query Q; that is, $r \setminus Q$. This ensures that the IDF value of the query term as well as other common words that are not stop words is negligible in the similarity computation, and guarantees that the agreement computation is not affected by this. Instead of normalizing the TF-IDF similarity by the normalization factor, we divide the TF-IDF similarity only by the highest TF value. Normalization was a necessity on the web, where web pages have no length limit. However, in the case of Twitter, the document size is bound (140 characters). Hence we do not penalize usage of the entire 140 characters, as they might bring in more content relevant to the query. However, we penalize tweets that repeat terms multiple times, as this does not increase the agreement value.

Agreement computation using POS weighted TF-IDF similarity may produce false positives if a pair of tweets is syntactically similar but semantically distinct. There may be false negatives too, for a pair of tweets that are syntactically different but semantically the same. Although our preliminary experiments show that the occurrence of these false negatives is minimal a more computationally expensive method such as Paraphrase Detection [22] or agreement computation considering synonyms from Wordnet may be considered.

4. RANKING

Our ranking of the candidate set R_Q needs to be sensitive to: (1) relevance of a specific result $r \in R_Q$ to Q; and (2) the trust reposed in r. These two (at times orthogonal) metrics must be combined into a single *score* for each r, in order to enable the ranking process.

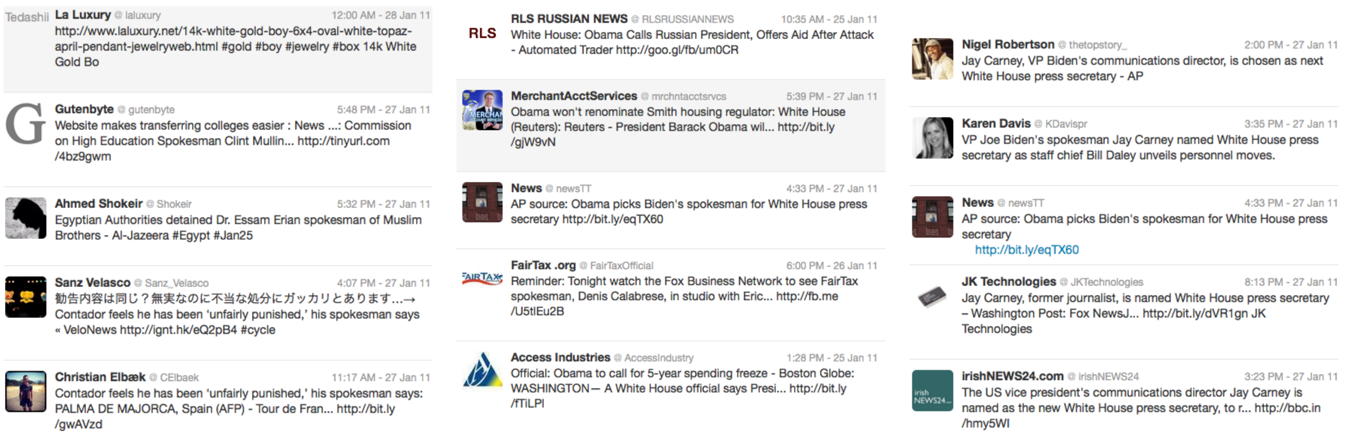
4.1 Agreement Graph

Computation of pairwise agreement between a pair of tweets represents the similarity of their content to each other, not to the query Q. Tweets which have low relevance to the query term may form high agreement cliques between themselves. This problem is well known in other fields as well, for example with PageRank [3] on the web. Hence we cannot use Agreement or Feature Score by themselves to compute a trustworthy and relevant Result Set. Instead, we use the agreement between tweets to construct clusters (and an agreement graph), and propagate the Feature Score over this. A more trustworthy cluster is expected to contain more tweets with a higher Feature Score.

Our candidate result set R_Q (for a specific query Q) is constructed such that all the tweets $t \in R_Q$ already bear a primary relevance to Q – tweets are chosen for inclusion in R_Q if they contain one or more keywords from the query, Q. We propagate the Feature Score on the agreement graph that is formed by the agreement analysis detailed above. This ensures that if there is a tweet in R_Q that is highly relevant to Q, it will not be suppressed simply because it did not have high enough Feature Score. More formally, we claim that the Feature Score of a tweet $t \in R_Q$ will be the sum of its current Feature Score and the Feature Score of all tweets that agree with t weighted by the magnitude of their agreement, i.e.

$$S'(Q, t_i) = S(Q, t_i) + \sum_{j \in E} w_{ij} \times S(Q, t_j) \;\forall\; (i, j) \in E$$

where w_j is the agreement between tweet t_i and t_j, and E is the set of edges in the agreement graph. The result set



Tedashii **La Luxury** @ laluxury 12:00 AM - 28 Jan 11
http://www.laluxury.net/14k-white-gold-boy-6x4-oval-white-topaz-april-pendant-jewelryweb.html #gold #boy #jewelry #box 14k White Gold Bo

Gutenbyte @ gutenbyte 5:48 PM - 27 Jan 11
Website makes transferring colleges easier : News ...: Commission on High Education Spokesman Clint Mullin... http://tinyurl.com/4bz9gwm

Ahmed Shokeir @ Shokeir 5:32 PM - 27 Jan 11
Egyptian Authorities detained Dr. Essam Erian spokesman of Muslim Brothers - Al-Jazeera #Egypt #Jan25

Sanz Velasco @ Sanz_Velasco 4:07 PM - 27 Jan 11
勧告内容は同じ？無実なのに不当な処分にガッカリとあります…→ Contador feels he has been 'unfairly punished,' his spokesman says « VeloNews http://ignt.hk/eQ2pB4 #cycle

Christian Elbæk @ CElbaek 11:17 AM - 27 Jan 11
Contador feels he has been 'unfairly punished,' his spokesman says: PALMA DE MAJORCA, Spain (AFP) - Tour de Fran... http://bit.ly/gwAVzd

(a) *Twitter Search.*

RLS RUSSIAN NEWS @ RLSRUSSIANNEWS 10:35 AM - 25 Jan 11
White House: Obama Calls Russian President, Offers Aid After Attack - Automated Trader http://goo.gl/fb/um0CR

MerchantAcctServices @ mrchntacctsrvcs 5:39 PM - 27 Jan 11
Obama won't renominate Smith housing regulator: White House (Reuters): Reuters - President Barack Obama wil... http://bit.ly/gjW9vN

News @ newsTT 4:33 PM - 27 Jan 11
AP source: Obama picks Biden's spokesman for White House press secretary http://bit.ly/eqTX60

FairTax .org @ FairTaxOfficial 6:00 PM - 26 Jan 11
Reminder: Tonight watch the Fox Business Network to see FairTax spokesman, Denis Calabrese, in studio with Eric... http://fb.me/U5tlEu2B

Access Industries @ AccessIndustry 1:28 PM - 25 Jan 11
Official: Obama to call for 5-year spending freeze - Boston Globe: WASHINGTON— A White House official says Presi... http://bit.ly/fTiLPI

(b) *Feature Score(FS)*

Nigel Robertson @ thetopstory_ 2:00 PM - 27 Jan 11
Jay Carney, VP Biden's communications director, is chosen as next White House press secretary - AP

Karen Davis @ KDavispr 3:35 PM - 27 Jan 11
VP Joe Biden's spokesman Jay Carney named White House press secretary as staff chief Bill Daley unveils personnel moves.

News @ newsTT 4:33 PM - 27 Jan 11
AP source: Obama picks Biden's spokesman for White House press secretary
http://bit.ly/eqTX60

JK Technologies @ JKTechnologies 8:13 PM - 27 Jan 11
Jay Carney, former journalist, is named White House press secretary – Washington Post: Fox NewsJ... http://bit.ly/dVR1gn JK Technologies

irishNEWS24.com @ irishNEWS24 3:23 PM - 27 Jan 11
The US vice president's communications director Jay Carney is named as the new White House press secretary, to r... http://bbc.in/hmy5WI

(c) *RAProp*

Figure 3: Top-5 ranked results for the query "White House spokesman replaced"

R_Q is ranked by the newly computed $S'(Q,t)$. The tweets are ranked based on the Feature Score computed after the propagation. The propagated Feature Scores may also be seen as weighted voting of other tweets that talk about the same content. The votes are weighted by their Feature Score since a vote from a highly trustworthy and popular tweet may be considered to be of higher value than a vote from an untrustworthy tweet.

Our method *RAProp*'s ranking for the query "`White House spokesman replaced`" is shown in Figure 3c. Although the tweets in the top-5 results are not from very popular users, these tweets are very relevant to the query as well as trustworthy in their content. The additional tweets that surfaced to the top-5 of the ranked results of *RAProp* had smaller Feature Scores before propagation. The top tweets from *RAProp* formed a tight cluster in the agreement graph due to the fact that there were a good number of tweets that were talking about the breaking news. Although the individual tweets do not have high Feature Score, the combined Feature Score of this cluster was higher than any other topic clusters formed for this query.

Using Feature Score weighted agreement helps us counter spam cluster voting. A tweet can be considered malicious either due to its content, or due to the content of links in that tweet. When the content is at fault, spam tweets will have high agreement with other spam tweets of the same content – the Feature Score of this entire cluster will be low. When the link is at fault instead, even though the content of the tweet might agree with other non-malicious tweets, the initial Feature Score will be penalized by the presence of the spam link. On the other hand, propagation helps us counter tweets from highly trustworthy users (and hence with high Feature Score) that may be untrustworthy [2]. This is simply because untrustworthy content is unlikely to find many independent endorsements. We evaluate the performance of our method, *RAProp*, in our experiments in Section 5.

4.2 Picking the Result Set R

For each query in our experiments, Q', we collect the top-K results returned by Twitter. These results become our initial candidate result set, R'. This set is then filtered to remove any re-tweets or replies, as the gold standard (TREC 2011 Microblog [20]) considers these tweets as irrelevant to the query. We add top-5 nouns terms that have the highest TF-IDF score to the query Q' to get the expanded query, Q. In order to constrain the expansion only to nouns, we run a Twitter NLP parser [10] to tag the tweets with parts of speech. The TF of each noun is then multiplied with its IDF value to compute the TF-IDF score. The top-N tweets returned by Twitter for the expanded query becomes the result set R.

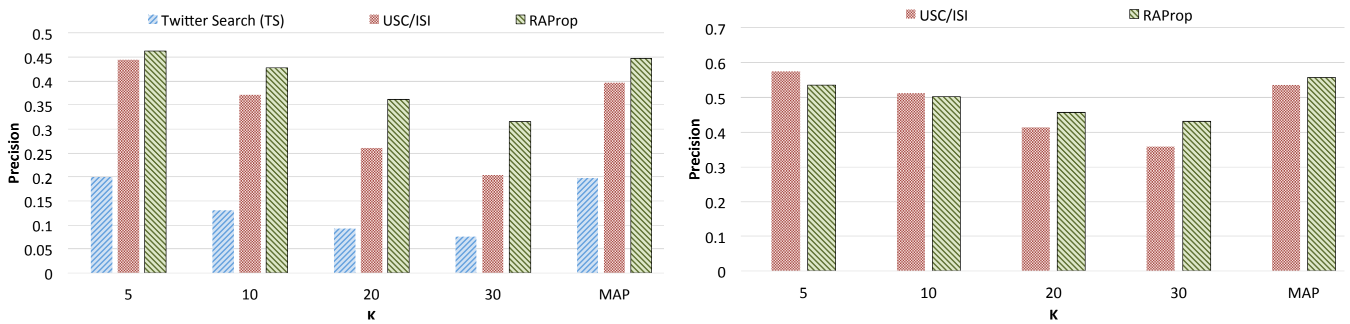
As mentioned in Section 3.2, IDF in twitter may not be able to prioritize the presence of nouns over the presence of a stop word. Hence, we compute the TF-IDF similarity of R by weighting the nouns higher (an order of 10) than other words. This is especially important in the case of Twitter as it contains spam tweets that try to match the stop words in the query in order to be part the results. We also remove tweets that contain less than 4 terms in them as these tweets mostly only contain the query terms and no other information. Twitter also matches query terms in URLs while returning results. Thus, we consider the URL words (with chunks split by special characters) as part of the tweet in order for agreement to account for keywords present in the URL alone. The tweets are stripped of punctuation, determiners and coordinating conjunctions so that agreement is only over the important terms.

5. EXPERIMENTS

In this section, we present an empirical evaluation of our proposed approach *RAProp*. We compare *RAProp* against Twitter search and the current best performing method on the TREC 2011 Microblog Dataset (USC/ISI [17]). We describe our experimental setup in Section 5.1, followed by the dataset used in Section 5.2. We then present results in Section 5.3.

5.1 Experimental Setup

Using the set of returned tweets R_Q that corresponds to a query Q, we evaluate each of the ranking methods. Since our dataset is offline (due to the use of the TREC dataset and the gold standard as described above), we have no direct way of running a Twitter search over that dataset. We thus simulate Twitter search (*TS*) on our dataset by sorting a copy of R_Q in reverse chronological order (i.e., latest first). We also use the current state of the art method (USC/ISI), as well as our proposed *RAProp* method, to rank R_Q. We set the bag size for our learning to rank method – Random Forest – as 10 and the maximum number of leaves for each tree as 20 to avoid over-fitting to the training data.



(a) *Against Twitter and USC/ISI while assuming a mediator model* (b) *Against USC/ISI on a non-mediator model*

Figure 4: External Evaluation of *RAProp*

We run our experiments in two different models: mediator model and non- mediator model. In mediator model, we assume that we do not own the Twitter Data and we access Twitter data only through a Twitter search API call. Hence the tweets in the candidate result set are the most recent N tweets that contain the one or more of the query terms. In non-mediator model, we assume that we store the entire Twitter data in-house and are not restricted by Twitter's relevance metric in selecting the result set R_Q. The mediator model is a more realistic scenario and it was adopted by TREC for their 2013 contest; however, we also compare non-mediator model performance of our method since previous iterations assumed such a model.

5.2 Dataset

For our evaluation, we used the TREC 2011 Microblog Dataset [20]. Our experiments were conducted on the 49 queries that are provided along with this dataset (and thus 49 different gold standards, one for each query, as defined previously). We used the Pagerank API in order to collect the PageRank of all the web URLs mentioned in the tweets in this set.

The TREC gold standard G_Q is a set of tweets annotated by TREC Microblog Track [20], where the annotations are with respect to their relevance to a given query Q. The relevance of each tweet is denoted by 3 discrete, mutually exclusive values $\{-1,\ 0,\ 1\}$:-1 stands for an untrustworthy tweet, 0 signifies irrelevance, and 1 stands for tweets that are relevant to the query. The gold standard gives us a way of evaluating tweets in the search results. It is generated by humans who examine the relevance of tweets to given queries. The gold standard may be considered as a measure of trustworthiness as well, as the tweets that are marked as untrustworthy (-1) are considered irrelevant to the query in our evaluations.

The maximum achievable precision in this dataset for 30 results($K = 30$) by re-ranking R_Q averaged over all 49 queries is 0.498 in the mediator model and 0.684 in the non-mediator model. Since we are interested in the relative performance of our method against the baselines, this is not a matter of concern.

5.3 Results

In this section, we evaluate the performance of our method *RAProp* against Twitter Search and USC/ISI [17]. USC/ISI uses a full dependence Markov Random Field model, Indri, to achieve a relevance score for each tweet in the dataset. Indri creates an off-line index on the entire tweets dataset in order to provide a relevance score for each tweet in the entire tweets dataset. This score along with other tweet specific features such as tweet length, existence of a URL or a hashtag is used by a Learning to Rank method to rank the tweets. In the non- mediator model, we run the queries over the entire tweet dataset index. On the mediator model, since we assume we do not have access to the entire dataset, we create a per-query index on the top-N tweets returned by twitter for that query.

As shown in Figure 4a, when we assume a mediator model, *RAProp* achieves higher precision for all values of K (10,20,30) than both current Twitter Search and USC/ISI method. When we compare the top-30 precision of *RAProp* against USC/ISI method and Twitter Search, we achieve a 53% and 300% improvement respectively. *RAProp* also achieves more than 125% and 13% higher MAP scores than Twitter search and USC/ISI method.

We also compare the precision of *RAProp* against USC/ISI method in a non-mediator model. Here USC/ISI method is able to index the entire tweet database. As shown in Figure 4b, the precision at K obtained by *RAProp* is equal to that of USC/ISI for K=10, with better results for higher values of K. *RAProp* is able to achieve a 20% higher top-30 precision than USC/ISI. Also, *RAProp* achieves a 4% higher MAP values than the USC/ISI ranking.

6. RELATED WORK

Although ranking tweets has received attention recently (c.f. [20, 17]), much of it has focused only on relevance. Most such approaches need background information on the query term which is usually not available for trending topics. A quality model based on the probability of re-tweeting [8] has been proposed which tries to associate the words in each tweet to the re-tweeting probability. We believe that the re-tweet probability of a tweet may not directly co-relate to the relevance of the tweet, since re-tweet probability of a tweet determines if the tweet is needed to be broadcast to the user's followers while relevance determines if the tweet is informative to the users. There are also multiple approaches [18, 15, 14] that try to rank tweets based on specific features of the user who tweeted the tweet. These methods are comparable to the Feature Score (*FS*) method. Our approach complements these, and can be seen as folding many of the features from previous work into a ranking algorithm. Ranking using the webpage mentioned as a part of the tweet has been considered [16].

The user follower-followee relation graph has been used to compute the popularity and trustworthy of a user [27, 1]. These approaches have no predictability when it comes to a user who is not part of the data set on which the popularity was found. They also take much longer to reflect changes in the relation graph into the the popularity score as the algorithm needs to be run over the entire follower-followee relation graph to get the new popularity values.

Credibility analysis of Twitter stories has been attempted by Castillo et al. [7, 12], who try to classify Twitter story threads as credible or non-credible. Our problem is different, since we try to assess the credibility of individual tweets. As the feature space is much smaller for an individual tweet – compared to a story thread – the problem becomes harder.

Agreement and propagation of trust over explicit links has been found to be effective in web scenarios [6, 13]. We cannot apply these directly to micro- blog scenarios as there are no explicit links between the documents. Finding relevant and trustworthy results based on implicit and explicit network structures has also been considered previously [11, 4]. To the best of our knowledge, ranking of tweets considering trust and content popularity has not been attempted. Ranking tweets based on the propagated user authority values have been attempted by Yang [28]. Since the propagation is done over the re-tweet graph, we expect tweets from popular users to be ranked higher. In contrast, we base our ranking also on the content and relevance to the query.

7. CONCLUSION

In this paper, we proposed *RAProp*, a microblog ranking mechanism for Twitter that combines two orthogonal features of trustworthiness– trustworthiness of source and trustworthiness of content, in order to filter out irrelevant results and spam. *RAProp* works by computing a *Feature Score* for each tweet and propagating that over a graph that represents content-based agreement between tweets, thus leveraging the collective intelligence embedded in tweets. Our detailed experiments [21] on a large TREC dataset showed that RAProp improves the precision of the returned results significantly over the baselines in both mediator and non-mediator models.

Acknowledgements

This research is supported by the ONR grants N000140910032, and N00014-13-1-0176, the NSF grant IIS-0905672 and two Google research awards.

8. REFERENCES

[1] M.-A. Abbasi and H. Liu. Measuring user credibility in social media. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 441–448. Springer, 2013.

[2] Twitter speaks, markets listen and fears rise. http://nyti.ms/ZuoSkj.

[3] R. Baeza-Yates, C. Castillo, V. López, and C. Telefónica. Pagerank increase under different collusion topologies. In *Proceedings of the 1st International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, pages 17–24, 2005.

[4] R. Balakrishnan and S. Kambhampati. Sourcerank: Relevance and trust assessment for deep web sources based on inter-source agreement. In *Proceedings of WWW*, 2011.

[5] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, pages 107–117, 1998.

[7] C. Castillo, M. Mendoza, and B. Poblete. Information credibility on twitter. In *Proceedings of WWW*, 2011.

[8] J. Choi, B. Croft, and J. K. Kim. Quality models for microblog retrieval. In *Proceedings of CIKM*, 2012.

[9] W. Cohen, P. Ravikumar, and S. Fienberg. A comparison of string distance metrics for name-matching tasks. In *Proceedings of IIWeb*, pages 73–78, 2003.

[10] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-speech tagging for twitter: annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics, 2011.

[11] M. Gupta and J. Han. Heterogeneous network-based trust analysis: a survey. *ACM SIGKDD Explorations*, pages 54–71, 2011.

[12] M. Gupta, P. Zhao, and J. Han. Evaluating event credibility on twitter. In *SMD*, 2012.

[13] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 576–587. VLDB Endowment, 2004.

[14] L. Jabeur, L. Tamine, and M. Boughanem. Featured tweet search: Modeling time and social influence for microblog retrieval. In *IEEE/WIC/ACM International Conference on Web Intelligence*, 2012.

[15] J. Jiang, L. Hidayah, T. Elsayed, and H. Ramadan. Best of kaust at trec-2011: Building effective search in twitter. In *Proceedings of the 20th Text REtrieval Conference (TREC 2011)*, 2012.

[16] R. McCreadie and C. Macdonald. Relevance in microblogs: Enhancing tweet retrieval using hyperlinked documents. 2012.

[17] D. Metzler and C. Cai. Usc/isi at trec 2011: Microblog track. In *Proceedings of the Text REtrieval Conference (TREC 2011)*, 2011.

[18] R. Nagmoti, A. Teredesai, and M. De Cock. Ranking approaches for microblog search. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, volume 1, pages 153 –157, 31 2010-sept. 3 2010.

[19] Twitter death hoaxes, alive and sadly, well. http://nyti.ms/10qVW9j.

[20] Trec 2011 microblog track. http://trec.nist.gov/data/tweets/.

[21] S. Ravikumar. *RAProp: Ranking Tweets by Exploiting the Tweet/User/Web Ecosystem.* PhD thesis, ARIZONA STATE UNIVERSITY, 2013.

[22] R. Socher, E. H. Huang, J. Pennington, A. Y. Ng, and C. D. Manning. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. *Advances in Neural Information Processing Systems*, 24:801–809, 2011.

[23] J. Teevan, D. Ramage, and M. R. Morris. #twittersearch: a comparison of microblog search and web search. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 35–44. ACM, 2011.

[24] Zombie followers and fake re-tweets. http://www.economist.com/node/21550333.

[25] State of twitter spam. http://bit.ly/d5PLDO.

[26] About top search results. http://bit.ly/IYssaa.

[27] Y. Yamaguchi, T. Takahashi, T. Amagasa, and H. Kitagawa. Turank: Twitter user ranking based on user-tweet graph analysis. In *Web Information Systems Engineering–WISE 2010*, pages 240–253. Springer, 2010.

[28] M. Yang, J. Lee, S. Lee, and H. Rim. Finding interesting posts in twitter based on retweet graph analysis. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1073–1074. ACM, 2012.