# Linking Tweets to News: A Framework to Enrich Short Text Data in Social Media

**Weiwei Guo**
Computer Science Department
Columbia University
weiwei@cs.columbia.edu

**Hao Li, Heng Ji**
Computer Science Department and Linguistic Department
Queens College and Graduate Center, City University of New York
{haoli.qc,hengjicuny}@gmail.com

**Mona Diab**
Computer Science Department
George Washington University
mtdiab@gwu.edu

## Abstract

Many current Natural Language Processing [NLP] techniques work well assuming a large context of text as input data. However they become ineffective when applied to short texts such as Twitter feeds. To overcome the issue, we want to find a related newswire document to a given tweet to provide contextual support for NLP tasks. This requires a robust modeling and understanding of the semantics of short text data tweets.

The contribution of the paper is two-fold: 1. we introduce the Linking-Tweets-to-News task as well as a dataset of linked tweet-news pairs, which can benefit many NLP applications; 2. in contrast to previous research which focuses on lexical features within the short texts (text-to-word information), we propose a graph based latent variable model that models the inter short text correlations (text-to-text information). This is motivated by the observation that a tweet usually only covers one aspect of an event. We show that using tweet specific feature (hashtag) and news specific feature (named entities) as well as temporal constraints, we are able to extract text-to-text correlations, and thus completes the semantic picture of a short text. Our experiments show significant improvement of our new model over baselines for three evaluation metrics in the new task.

## 1 Introduction

Recently there has been an increasing interest in language understanding of Twitter messages. Researchers (Speriosui et al., 2011; Brody and Diakopoulos, 2011; Jiang et al., 2011) are interested in sentiment analysis on Twitter feeds, and opinion mining towards political issues or politicians (Tumasjan et al., 2010; Conover et al., 2011). Others (Ramage et al., 2010; Jin et al., 2011) summarize tweets using topic models. Although these NLP techniques are mature, their performance on tweets inevitably degrades, due to the inherent sparsity in short texts. In the case of sentiment analysis, while people are able to achieve 87.5% accuracy (Maas et al., 2011) on a movie review dataset (created in (Pang and Lee, 2004)), the performance drops to 75% (Li et al., 2012) on a sentence level movie review dataset (created in (Pang and Lee, 2005)). The problem worsens when some existing NLP systems cannot produce any results given the short texts. Considering the following tweet:

*Pray for Mali...*

An event extraction/discovery system, e.g. (Ji and Grishman, 2008), fails to discover the *war* event due to the lack of context information (Benson et al., 2011), thus failing to shed light on the users focus/interests.

To enable the NLP tools to better understand Twitter feeds, we propose the task of linking a tweet to a news article that is relevant to the tweet, thereby augmenting the context of the tweet. In the above example, we want to supplement the implicit context of the above tweet with a news article such as the following entitled:

*State of emergency declared in Mali*

where abundant evidence can be fed into an off-the-shelf event extraction/discovery system. To create a gold standard dataset, we download tweets spanning over 18 days, each with a url linking to a news article of CNN or NYTIMES, as well as all the news of CNN and NYTIMES published in the period. The goal is to predict the url referred news article based on the text in each tweet.[1] We believe

[1]The data and code is publicly available at www.cs.

many NLP tasks will benefit from this task. In fact, in the topic model research, previous work (Jin et al., 2011) already showed that by incorporating webpages whose urls are contained in tweets, the tweet clustering purity score is boosted from $0.280$ to $0.392$.

Given the few number of words in a tweet (14 words on average in our dataset), the traditional high dimensional surface word matching is lossy and fails to pinpoint the news article. This constitutes a classic short text semantics impediment (Agirre et al., 2012). Latent variable models are powerful by going beyond the surface word level and mapping short texts into a low dimensional dense vector (Socher et al., 2011; Guo and Diab, 2012a). Accordingly, we apply a latent variable model, namely, the Weighted Textual Matrix Factorization [WTMF] (Guo and Diab, 2012a; Guo and Diab, 2012b) to both the tweets and the news articles. WTMF is a state-of-the-art unsupervised model that was tested on two short text similarity datasets: (Li et al., 2006) and (Agirre et al., 2012), which outperforms Latent Semantic Analysis [LSA] (Landauer et al., 1998) and Latent Dirichelet Allocation [LDA] (Blei et al., 2003) by a large margin. We employ it as a strong baseline in this task as it exploits and effectively models the missing words information in a tweet, in practice adding thousands of more features for the tweet, by contrast LDA, for example, only leverages observed words (14 features) to infer the latent vector for a tweet.

Apart from the data sparseness, our dataset proposes another challenge: a tweet usually covers only one aspect of an event. In our previous example, the tweet only contains the location *Mali* while the event is about French army participated in Mali war. In this scenario, we would like to find the missing elements of the tweet such as *French, war* from other short texts, to complete the semantic picture of *Pray in Mali* tweet. One drawback of WTMF for our purposes is that it simply models the text-to-word information without leveraging the correlation between short texts. While this is acceptable on standard short text similarity datasets (data points are independently generated), it ignores some valuable information characteristically present in our dataset: (1) The tweet specific features such as hashtags. Hashtags prove to be a direct indication of the semantics of tweets (Ramage et al., 2010); (2) The news specific features such as named entities in a document. Named entities acquired from a news document, typically with high accuracy using Named Entity Recognition [NER] tools, may be particularly informative. If two texts mention the same entities then they might describe the same event; (3) The temporal information in both data (tweets and news articles). We note that there is a higher chance of event description overlap between two texts if their time of publication is similar.

In this paper, we study the problem of mining and exploiting correlations between texts using these features. Two texts may be considered related or complementary if they share a hashtag/NE or satisfies the temporal constraints. Our proposed latent variable model not only models text-to-word information, but also is aware of the text-to-text information (illustrated in Figure 1): two linked texts should have similar latent vectors, accordingly the semantic picture of a tweet is completed by receiving semantics from its related tweets. We incorporate this additional information in the WTMF model. We also show the different impact of the text-to-text relations in the tweet genre and news genre. We are able to achieve significantly better results than with a text-to-words WTMF model. This work can be regarded as a short text modeling approach that extends previous work however with a focus on combining the mining of information within short texts coupled with utilizing extra shared information across the short texts.

## 2 Task and Data

The task is given the text in a tweet, the system aims to find the most relevant news article. For gold standard data, we harvest all the tweets that have a single url link to a CNN or NYTIMES news article, dated from the 11th of Jan to the 27th of Jan, 2013. In evaluation, we consider this url-referred news article as the gold standard – the most relevant document for the tweet, and remove the url from the text of the tweet. We also collect all the news articles from both CNN and NYTIMES from RSS feeds during the same timeframe. Each tweet entry has the published time, author, text; each news entry contains published time, title, news summary, url. The tweet/news pairs are extracted by matching urls. We manually filtered "trivial" tweets where the tweet content is

columbia.edu/~weiwei

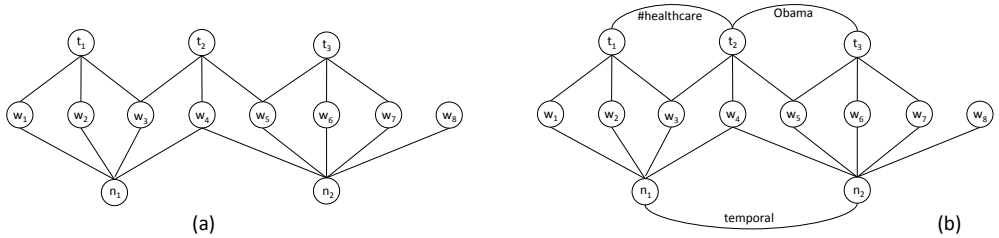



Figure 1: (a) WTMF. (b) WTMF-G: the tweet nodes $t$ and news nodes $n$ are connected by hashtags, named entities or temporal edges (for simplicity, the missing tokens are not shown in the figure)

simply the news title or news summary. The final dataset results in 34,888 tweets and 12,704 news articles.

It is worth noting that the news corpus is not restricted to current events. It covers various genres and topics, such as travel guides. e.g. *World's most beautiful lakes*, and health issues, e.g. *The importance of a 'stop day'*, etc.

### 2.1 Evaluation metric

For our task evaluation, ideally, we would like the system to be able to identify the news article specifically referred to by the url within each tweet in the gold standard. However, this is very difficult given the large number of potential candidates, especially those with slight variations. Therefore, following the Concept Definition Retrieval task in (Guo and Diab, 2012a) and (Steck, 2010) we use a metric for evaluating the ranking of the correct news article to evaluate the systems, namely, ATOP$_t$, *area under the* TOPK$_t(k)$ *recall curve for a tweet* $t$. Basically, it is the normalized ranking $\in [0, 1]$ of the correct news article among all candidate news articles: ATOP$_t = 1$ means the url-referred news article has the highest similarity value with the tweet (a correct NARU); ATOP$_t = 0.95$ means the similarity value with correct news article is larger than $95\%$ of the candidates, i.e. within the top $5\%$ of the candidates. ATOP$_t$ is calculated as follows:

$$\text{ATOP}_t = \int_0^1 \text{TOPK}_t(k)\text{d}k \qquad (1)$$

where TOPK$_t(k) = 1$ if the url referred news article is in the "top $k$" list, otherwise TOPK$_t(k) = 0$. Here $k \in [0, 1]$ is the relative position (when $k = 1$, it means all the candidates).

We also include other metrics to examine if the system is able to rank the url referred news article in the first few returned results: **TOP10** recall hit rate to evaluate whether the correct news is in the top 10 results, and **RR**, Reciprocal Rank$= 1/r$ (i.e., RR$= 1/3$ when the correct news article is ranked at the 3rd highest place).

## 3 Weighted Textual Matrix Factorization

The WTMF model (Guo and Diab, 2012a) has been successfully applied to the short text similarity task, achieving state-of-the-art unsupervised performance. This can be attributed to the fact that it models the missing tokens as features, thereby adding many more features for a short text. The missing words of a sentence are defined as all the vocabulary of the training corpus minus the observed words in a sentence. Missing words serve as negative examples for the semantics of a short text: the short text should not be related to the missing words.

As per (Guo and Diab, 2012a), the corpus is represented in a matrix $X$, where each cell stores the TF-IDF values of words. The rows of $X$ are words and columns are short texts. As in Figure 2, matrix $X$ is approximated by the product of a $K \times M$ matrix $P$ and a $K \times N$ matrix $Q$. Accordingly, each sentence $s_j$ is represented by a $K$ dimensional latent vector $Q_{\cdot,j}$. Similarly a word $w_i$ is generalized by $P_{\cdot,i}$. Therefore, the inner product of a word vector $P_{\cdot,i}$ and a short text vector $Q_{\cdot,j}$ is to approximate the cell $X_{ij}$ (shaded part in Figure 2). In this way, the missing words are modeled by requiring the inner product of a word vector and short text vector to be close to 0 (the word and the short text should be irrelevant).

Since 99% cells in $X$ are missing tokens (0 value), the impact of observed words is significantly diminished. Therefore a small weight $w_m$ is assigned for each 0 cell (missing tokens) in the matrix $X$ in order to preserve the influence of observed words. $P$ and $Q$ is optimized by minimize the objective function:

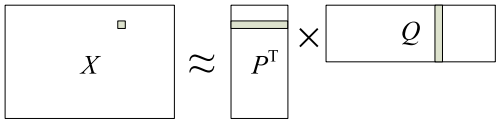



Figure 2: Weighted Textual Matrix Factorization

$$\sum_i \sum_j W_{ij} \left(P_{\cdot,i} \cdot Q_{\cdot,j} - X_{ij}\right)^2 + \lambda ||P||_2^2 + \lambda ||Q||_2^2$$

$$W_{i,j} = \begin{cases} 1, & \text{if } X_{ij} \neq 0 \\ w_m, & \text{if } X_{ij} = 0 \end{cases} \quad (2)$$

where $\lambda$ is a regularization term.

## 4 Creating Text-to-text Relations via Twitter/News Features

WTMF exploits the text-to-word information in a very nuanced way, while the dependency between texts is ignored. In this Section, we introduce how to create text-to-text relations.

### 4.1 Hashtags and Named Entities

Hashtags highlight the topics in tweets, e.g., *The #flu season has started*. We believe two tweets sharing the same hashtag should be related, hence we place a link between them to explicitly inform the model that these two tweets should be similar.

We find only 8,701 tweets out of 34,888 has a hashtag. In fact, we observe many hashtag words mentioned in tweets without explicitly being tagged with #. To overcome the hashtag sparseness issue, one can resort to keywords recommendation algorithms to mine hashtags for the tweets (Yang et al., 2012). In this paper, we adopt a simple but effective approach: we collect all the hashtags in the dataset, and automatically hashtag any word in a tweet if that word appears hashtagged in any other tweet. This process resulted in 33,242 tweets automatically labeled with hashtags. For each tweet, and for each hashtag it contains, we extract $k$ tweets that contain this hashtag, assuming them as tweets that are complementary to the target tweet, and link the $k$ tweets to the target tweet. If there are more than $k$ tweets found, we choose $k$ ones that are most chronologically close to the target tweet. The statistics of links can be found in table 2.

Named entities are some of the most salient features in a news article. Directly applying NER tools on news titles or tweets results in many errors (Liu et al., 2011) due to the noise in the data, e.g. slang and capitalization. Accordingly, we first apply the NER tool on news summaries, then label named entities in the tweets in the same way as labeling the hashtags: if there is a string in the tweet that matches a named entity from the summaries, then it is labeled as a named entity in the tweet. 25,132 tweets are assigned at least one named entity.[2] To create the similar tweet set, we extract $k$ tweets for each named entity in a tweet.

### 4.2 Temporal Relations

Tweets published in the same time interval has a larger chance of being similar than those are not chronologically close (Wang and McCallum, 2006). However, we cannot simply assume any two tweets are similar only based on the timestamp. Therefore, for a tweet we link it to the $k$ most similar tweets whose published time is within 24 hours of the target tweet's timestamp. We use the similarity score returned by WTMF model to measure the similarity of two tweets.

We experimented with other features such as authorship. We note that it was not a helpful feature. While authorship information helps in the task of news/tweets recommendation for a *user* (Corso et al., 2005; Yan et al., 2012), it is too general for this task where we target on "recommending" a news article for a *tweet*.

### 4.3 Creating Relations on News

We extract the 3 subgraphs (based on hashtags, named entities and temporal) on news articles. However, automatically tagging hashtags or named entities leads to much worse performance (around 93% ATOP values, a 3% decrease from baseline WTMF). There are several reasons for this: 1. When a hashtag-matched word appears in a tweet, it is often related to the central meaning of the tweet, however news articles are generally much longer than tweets, resulting in many more hashtags/named entities matches even though these named entities may not be closely related. 2. The noise introduced during automatic named entity tagging accumulates much faster given the large number of named entities in news data. Therefore we only extract temporal relations for news articles.

[2]Note that there are some false positive named entities detected such as *apple*. We plan to address removing noisy named entities and hashtags in future work

## 5 WTMF on Graphs

We propose a novel model to incorporate the links generated as described in the previous section.

If two texts are connected by a link, it means they should be semantically similar. In the WTMF model, we would like the latent vectors of two text nodes $Q_{\cdot,j_1}, Q_{\cdot,j_2}$ to be as similar as possible, namely that their cosine similarity to be close to 1. To implement this, we add a regularization term in the objective function of WTMF (equation 2) for each linked pairs $Q_{\cdot,j_1}, Q_{\cdot,j_2}$:

$$\delta \cdot (\frac{Q_{\cdot,j_1} \cdot Q_{\cdot,j_2}}{|Q_{\cdot,j_1}||Q_{\cdot,j_2}|} - 1)^2 \quad (3)$$

where $|Q_{\cdot,j}|$ denotes the length of vector $Q_{\cdot,j}$. The coefficient $\delta$ denotes the importance of the text-to-text links. A larger $\delta$ means we put more weight on the text-to-text links and less on the text-to-word links. We refer to this model as WTMF-G (WTMF on graphs).

### 5.1 Inference

Alternating Least Square [ALS] is used for inference in weighted matrix factorization (Srebro and Jaakkola, 2003). However, ALS is no longer applicable here with the new regularization term (equation 3) involving the length of text vectors $|Q_{\cdot,j}|$, which is not in quadratic form. Therefore we approximate the objective function by treating the vector length $|Q_{\cdot,j}|$ as fixed values during the ALS iterations:

$$\begin{aligned} P_{\cdot,i} &= \left(Q\tilde{W}^{(i)}Q^\top + \lambda I\right)^{-1} Q\tilde{W}^{(i)}X_{\cdot,i} \\ Q_{\cdot,j} &= \left(P\tilde{W}^{(j)}P^\top + \lambda I + \delta L_{(j)}^2 Q_{\cdot,s(j)}\text{diag}(L_{(s(j))}^2)Q_{\cdot,s(j)}^\top\right)^{-1} \\ &\quad \left(P\tilde{W}^{(j)}X_{j,\cdot}^\top + \delta L_{(j)}Q_{\cdot,s(j)}L_{n(j)}\right) \end{aligned} \quad (4)$$

We define $n(j)$ as the linked neighbors of short text $j$, and $Q_{\cdot,n(j)}$ are the set of latent vectors of $j$'s neighbors. The reciprocal of length of these vectors in the current iteration are stored in $L_{s(j)}$. Similarly, the reciprocal of the length of the short text vector $Q_{\cdot,j}$ is $L_j$. $\tilde{W}^{(i)} = \text{diag}(W_{\cdot,i})$ is an $M \times M$ diagonal matrix containing the $i$th row of weight matrix $W$. Due to limited space, the details of the optimization are not shown in this paper; they can be found in (Steck, 2010).

## 6 Experiments

### 6.1 Experiment Setting

**Corpora:** We use the same corpora as in (Guo and Diab, 2012a): Brown corpus (each sentence is treated as a document), sense definitions of Wiktionary and Wordnet (Fellbaum, 1998). The tweets and news articles are also included in the corpus, generating 441,258 short texts and 5,149,122 words. The data is tokenized, POS-tagged by Stanford POS tagger (Toutanova et al., 2003), and lemmatized by WordNet::QueryData.pm. The value of each word in matrix $X$ is its TF-IDF value in the short text.

**Baselines:** We present 4 baselines: 1. Information Retrieval model [IR], which simply treats a tweet as a document, and performs traditional surface word matching. 2. LDA-$\theta$ with Gibbs Sampling as inference method. We use the inferred topic distribution $\theta$ as a latent vector to represent the tweet/news. 3. LDA-$wvec$. The problem with LDA-$\theta$ is the inferred topic distribution latent vector is very sparse with only a few non-zero values, resulting in many tweet/news pairs receiving a high similarity value as long as they are in the same topic domain. Hence following (Guo and Diab, 2012a), we first compute the latent vector of a word by $P(z|w)$ (topic distribution per word), then average the word latent vectors weighted by TF-IDF values to represent the short text, which yields much better results. 4. WTMF. In these baselines, hashtags and named entities are simply treated as words.

To curtail variation in results due to randomness, each reported number is the average of 10 runs. For WTMF and WTMF-G, we assign the same initial random values and run 20 iterations. In both systems we fix the missing words weight as $w_m = 0.01$ and regularization coefficient at $\lambda = 20$, which is the best condition of WTMF found in (Guo and Diab, 2012a; Guo and Diab, 2012b). For LDA-$\theta$ and LDA-$wvec$, we run Gibbs Sampling based LDA for 2000 iterations and average the model over the last 10 iterations.

**Evaluation:** The similarity between a tweet and a news article is measured by cosine similarity. A news article is represented as the concatenation of its title and its summary, which yields better performance.[3]

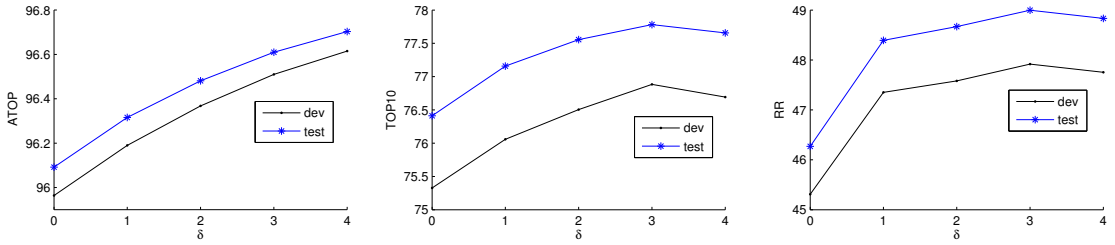
As in (Guo and Diab, 2012a), for each tweet, we collect the 1,000 news articles published prior to the tweet whose dates of publication are closest to that of the tweet. [4] The cosine similarity

[3] While these are separated, WTMF receive ATOP $95.558\%$ for representing news article as titles and $94.385\%$ for representing news article as summaries

[4] Ideally we want to include all the news articles published

| Models | Parameters | ATOP | | TOP10 | | RR | |
|---|---|---|---|---|---|---|---|
| | | dev | test | dev | test | dev | test |
| IR | - | 90.795% | 90.743% | 73.478% | 74.103% | 46.024% | 46.281% |
| LDA-$\theta$ | $\alpha = 0.05, \beta = 0.05$ | 81.368% | 81.251% | 32.328% | 31.207% | 13.134% | 12.469% |
| LDA-$wvec$ | $\alpha = 0.05, \beta = 0.05$ | 94.148% | 94.196% | 53.500% | 53.952% | 28.743% | 27.904% |
| WTMF | - | 95.964% | 96.092% | 75.327% | 76.411% | 45.310% | 46.270% |
| WTMF-G | $k = 3, \delta = 3$ | 96.450% | 96.543% | 76.485% | 77.479% | 47.516% | 48.665% |
| WTMF-G | $k = 5, \delta = 3$ | **96.613%** | **96.701%** | 76.029% | 77.176% | 47.197% | 48.189% |
| WTMF-G | $k = 4, \delta = 3$ | 96.510% | 96.610% | **77.782%** | **77.782%** | **47.917%** | **48.997%** |

Table 1: ATOP Performance (latent dimension $D = 100$ for LDA/WTMF/WTMF-G)




(a) ATOP (b) TOP10 (c) RR

Figure 3: Impact of $\delta$ ($D = 100, k = 4$)

score between the url referred news article and the tweet is compared against the scores of these 1,000 news articles to calculate the metric scores. 1/10 of the tweet/news pairs are used as development set, based on which all the parameters are tuned. The metrics ATOP, TOP10 and RR are used to evaluate the performance of systems.

### 6.2 Results

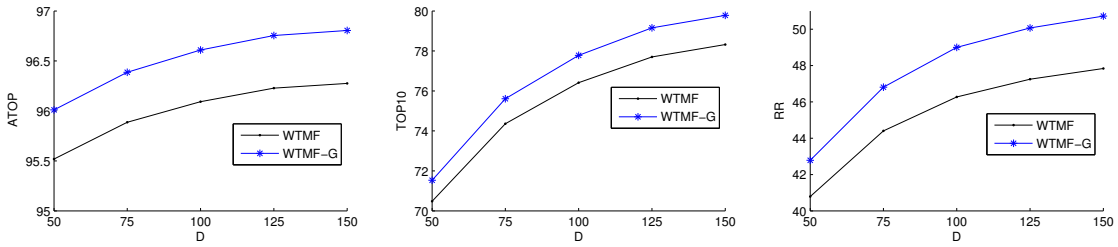
Table 1 summarizes the performance of the baselines and WTMF-G at latent dimension $D = 100$. All the parameters are chosen based on the development set. For WTMF-G, we try different values of $k$ (the number of neighbors linked to a tweet/news for a hashtag/NE/time constraint) and $\delta$ (the weight of link information). We choose to model the links in four subgraphs: (a) hashtags on tweet; (b) named entities on tweet; (c) time on tweet; (d) time on news article. For LDA we tune the hyperparameter $\alpha$ (Dirichlet prior for topic distribution of a document) and $\beta$ (Dirichlet prior for word distribution given a topic). It is worth noting that ATOP measures the overall ranking in 1000 samples while TOP10/RR focus on whether the aligned news article is in the first few returned results.

Same as reported in (Guo and Diab, 2012a), LDA-$\theta$ has the worst results due to directly using the inferred topic distribution of a text $\theta$. The inferred topic vector has only a few non-zero values, hence a lot of information is missing. LDA-$wvec$ preserves more information by creating a dense latent vector from the topic distribution of a word $P(z|w)$, and thus does much better in ATOP.

It is interesting to see that IR model has a very low ATOP (90.795%) and an acceptable RR (46.281%) score, in contrast to LDA-$wvec$ with a high ATOP (94.148%) and a low RR(27.904%) score. This is caused by the nature of the two models. LDA-$wvec$ is able to identify global coarse grained topic information (such as *politics* vs. *economics*), hence receiving a high ATOP by excluding most irrelevant news articles, however it does not distinguish fine grained difference such as *Hillary* vs. *Obama*. IR model exerts the opposite influence via word matching. It ranks a correct news article very high if overlapping words exist (leading to a high RR), or the news article is ranked very low if no overlapping words (hence a low ATOP).

We can conclude WTMF is a very strong baseline given that it achieves high scores in 3 metrics. As a latent variable model, it is able to capture global topics (+1.89% ATOP over LDA-$wvec$); moreover, by explicitly modeling missing words, the existence of a word is also encoded in the latent vector (+2.31% TOP10 and −0.011% RR over IR model).

prior to the tweet, however, that will give a bias to some tweets, since the latter tweets have a larger candidate set than the earlier ones




(a) ATOP (b) TOP10 (c) RR

Figure 4: Impact of latent dimension $D$ ($k = 4$)

| Conditions | Links | ATOP | | TOP10 | | RR | |
|---|---|---|---|---|---|---|---|
| | | dev | test | dev | test | dev | test |
| hashtags_tweets | 375,371 | +0.397% | +0.379% | +1.015% | +1.021% | +0.504% | +0.641% |
| NE_tweets | 164,412 | +0.141% | +0.130% | +0.598% | +0.479% | +0.278% | +0.294% |
| time_tweet | 139,488 | +0.126% | +0.136% | +0.512% | +0.503% | +0.241% | +0.327% |
| time_news | 50,008 | +0.036% | +0.026% | +0.156% | +0.256% | +1.890% | +1.924% |
| full model (all 4 subgraphs) | 573,999 | **+0.546%** | **+0.518%** | **+1.556%** | +1.371% | **+2.607%** | **+2.727%** |
| full model **minus** hashtags_tweets | 336,963 | +0.288% | +0.276% | +1.129% | +1.037% | +2.488% | +2.541% |
| full model **minus** NE_tweets | 536,333 | +0.528% | +0.503% | +1.518% | **+1.393%** | +2.580% | +2.680% |
| full model **minus** time_tweet | 466,207 | +0.457% | +0.426% | +1.281% | +1.145% | +2.449% | +2.554% |
| full model **minus** time_news | 523,991 | +0.508% | +0.490% | +1.300% | +1.190% | +0.632% | +0.785% |
| author_tweet | 21,318 | +0.043% | +0.042% | +0.028% | +0.057% | −0.003% | −0.017% |
| full model **plus** author_tweet | 593,483 | +0.575% | +0.545% | +1.465% | +1.336% | +2.415% | +2.547% |

Table 2: Contribution of subgraphs when $D = 100, k = 4, \delta = 3$ (gain over baseline WTMF)

With WTMF being a very challenging baseline, WTMF-G can still significantly improve all 3 metrics. In the case $k = 4, \delta = 3$ compared to WTMF, WTMF-G receives $+1.371\%$ TOP10, $+2.727\%$ RR, and $+0.518\%$ ATOP value (this is a significant improvement of ATOP value considering that it is averaged on 30,000 data points, at an already high level of $96\%$ reducing error rate by $13\%$). All the improvement of WTMF-G over WTMF is statistically signicant at the 99% condence level with a two-tailed paired t-test.

We also present results using different number of links $k$ in WTMF-G. We experiment with $k = \{3, 4, 5\}$. $k = 4$ is found to be the optimal value (although $k = 5$ has a better ATOP). Figure 3 demonstrates the impact of $\delta = \{0, 1, 2, 3, 4\}$ on each metric when $k = 4$. Note when $\delta = 0$ no link is used, which is the baseline WTMF. We can see using links is always helpful. When $\delta = 4$, we receive a higher ATOP value but lower TOP10 and RR.

Figure 4 illustrates the impact of dimension $D = \{50, 75, 100, 125, 150\}$ on WTMF and WTMF-G ($k = 4$) on the test set. In the case $D = 125$ and $D = 150$, we select a different $\delta = 0.7$ based on development set. This is because the length of the latent vectors become longer with more dimensions, and a longer length implies more weight for links, (according to equation 3). The trends hold in different $D$ values with a consistent improvement. Generally a larger $D$ leads to a better performance. In all conditions WTMF-G outperforms WTMF.

### 6.3 Contribution of Subgraphs

We are interested in the contribution of each feature subgraph. Therefore we list the impact of individual components in table 2. The impact of each subgraph is evaluated in two conditions: (a) the *subgraph-only*; (b) the *full-model-minus* the subgraph. The *full model* is the combination of the 4 subgraphs (which is also the best model $k = 4$ in table 1). In the last two rows of table 2 we also present the results of using authorship only and the full model plus authorship. The 2nd column lists the number of links in the subgraph. To highlight the difference, we report the gain of each model over the baseline model WTMF.

We have several interesting observations from table 2. It is clear that the hashtag subgraph on tweets is the most useful subgraph: with hashtag_tweet it has the best ATOP and TOP10 values among *subgraph-only* condition (ATOP: $+0.379\%$ vs. 2nd best $+0.136\%$, TOP10: $+1.021\%$ vs. 2nd best $+0.503\%$), while in the full-model-minus condition, minus hashtag has the lowest ATOP and TOP10. Observing that it also contains the most links, we believe the cover-

age is also an important reason for the great performance.

It seems the named entity subgraph helps the least. Looking into the extracted named entities and hashtags, we find many popular named entities are covered by hashtags. Nevertheless, adding named entity subgraph into final model has a positive contribution to ATOP and TOP10.

It is also worth noting that the time_news subgraph has the most positive influence in RR. This is because temporal information is very salient in news domain: usually there are several reports to describe an event within a short period, therefore the news latent vector is strengthened by receiving semantics from its neighbors.

At last, we analyze the influence of authorship of tweets. Adding authorship into the full model greatly hurts the score of TOP10 and RR, whereas it is helpful to ATOP. This is understandable since by introducing author link between tweets, we are actually averaging the latent vectors of tweets written by the same person. Therefore, for a tweet whose topic is vague, it will get some prior knowledge of topics through the author links (hence increase ATOP), whereas this knowledge becomes noise for the tweets that are already handled very well (hence decrease TOP10 and RR).

### 6.4 Error Analysis

We look closely into ATOP results to obtain an intuitive feel for what is captured and what is not. For example, the ATOP score of WTMF for the tweet-news pair below is 89.9%:

Tweet: *...stoked growing speculation that Pakistan's powerful military was quietly supporting moves... @declanwalsh*
News: *Pakistan Supreme Court Orders Arrest of Prime Minister*

By identifying "Pakistan" and "Supreme Court" as hashtags/named entity, WTMF-G is able to propagate the semantics from the two following informative tweets to the original tweet, hence achieving a higher ATOP score of 91.9%.

*#Pakistan Supreme Court orders the arrest of the PM on corruption charges.*
*A discouraging sign from a tumultuous political system: Pakistan's Supreme Court ordered the arrest of PM Ashraf today.*

Below is an example that shows the deficiency of both WTMF and WTMF-G:
Tweet: *Another reason to contemplate moving: an early death*
News: *America flunks its health exam*

In this case WTMF and WTMF-G achieves a low ATOP of $69.8\%$ and $75.1\%$, respectively. The only evidence the latent variable models rely on is only lexical items (WTMF-G extract additional text-to-text correlation by word matching). To pinpoint the url referred news article, other advanced NLP features should be exploited. In this case, we believe sentiment information could be helpful – both tweet and the news article contains a negative polarity.

## 7 Related Work

**Short Text Semantics:** The field of short text semantics has progressed immensely in recent years. Early work focus on word pair similarity in the high dimensional space. The word pair similarity is either knowledge based (Mihalcea et al., 2006; Tsatsaronis et al., 2010) or corpus based (Li et al., 2006; Islam and Inkpen, 2008), where co-occurrence information cannot be efficiently exploited. Guo and Diab (2012a) show the superiority of the latent space approach in the WTMF model achieving state-of-the-art performance on two datasets. However, all of them only reply on text-to-word information. In this paper, we focus on modeling inter-text relations induced by Twitter/news features. We extend the WTMF model and adapt it into tweets modeling, achieving significantly better results.

**Modeling Tweets in a Latent Space:** Ramage et al. (2010) also use hashtags to improve the latent representation of tweets in a LDA framework, Labeled-LDA (Ramage et al., 2009), treating each hashtag as a label. Similar to the experiments presented in this paper, the result of using Labeled-LDA alone is worse than the IR model, due to the sparseness in the induced LDA latent vector. Jin et al. (2011) apply an LDA based model on clustering by incorporating url referred documents. The semantics of long documents are transferred to the topic distribution of tweets.

**News recommendation:** A news recommendation system aims to recommend news articles to a user based on the features (e.g., key words, tags, category) in the documents that the user likes (hence these documents form a training set) (Claypool et al., 1999; Corso et al., 2005; Lee and Park, 2007). Our paper resembles it in searching for a related news article. However, we target on rec-

ommending news article only based on a tweet, which is a much smaller context than the set of favorite documents chosen by a user .

**Research on Tweets:** In (Duan et al., 2010), url availability is an important feature for tweets ranking. However, the number of tweets with an explicit url is very limited. Huang et al. (2012) propose a graph-based framework to propagate tweet ranking scores, where relevant web documents is found to be helpful to discover informative tweets. Both work can take advantage of our work to either extract potential url features or retrieve topically similar web documents.

## 8 Conclusion

We propose a Linking-Tweets-to-News task, which potentially benefits many NLP applications where off-the-shelf NLP tools can be applied to linked news. We also collect a gold standard dataset by crawling tweets with a url referring to a news. We formalize the linking task as a short text modeling problem, and extract Twitter/news specific features to extract text-to-text relations, which are incorporated in a latent variable model. We achieve significant improvement over baselines.

## Acknowledgements

This work was supported by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF- 09-2-0053 (NS-CTA), the U.S. NSF CAREER Award under Grant IIS-0953149, the U.S. NSF EAGER Award under Grant No. IIS-1144111, the U.S. DARPA FA8750-13-2-0041 - Deep Exploration and Filtering of Text (DEFT) Program and CUNY Junior Faculty Award. 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 U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

## References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *First Joint Conference on Lexical and Computational Semantics (*SEM)*.

Edward Benson, Aria Haghighi, and Regina Barzilay. 2011. Event discovery in social media feeds. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3.

Samuel Brody and Nicholas Diakopoulos. 2011. Cooooooooooooooollllllllllllll!!!!!!!!!!!!!! using word lengthening to detect sentiment in microblogs. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*.

Mark Claypool, Anuja Gokhale, Tim Miranda, Pavel Murnikov, Dmitry Netes, and Matthew Sartin. 1999. Combining content-based and collaborative filters in an online newspaper. In *In Proceedings of the ACM SIGIR Workshop on Recommender Systems*.

Michael Conover, Jacob Ratkiewicz, Matthew Francisco, Bruno Gonçalves, Filippo Menczer, and Alessandro Flammini. 2011. Political polarization on twitter. In *ICWSM*.

Gianna M. Del Corso, Antonio Gulli, and Francesco Romani. 2005. Ranking a stream of news. In *WWW*, pages 97–106.

Yajuan Duan, Long Jiang, Tao Qin, Ming Zhou, and Heung-Yeung Shum. 2010. An empirical study on learning to rank of tweets. In *COLING*.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Weiwei Guo and Mona Diab. 2012a. Modeling sentences in the latent space. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*.

Weiwei Guo and Mona Diab. 2012b. Weiwei: A simple unsupervised latent semantics based approach for sentence similarity. In *First Joint Conference on Lexical and Computational Semantics (*SEM)*.

Hongzhao Huang, Arkaitz Zubiaga, Heng Ji, Hongbo Deng, Dong Wang, Hieu Le, Tarek Abdelzather, Jiawei Han, Alice Leung, John Hancock, and Clare Voss. 2012. Tweet ranking based on heterogeneous networks. In *Proceedings of the 24th International Conference on Computational Linguistics*.

Aminul Islam and Diana Inkpen. 2008. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data*, 2.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL-08: HLT*.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of Association for Computational Linguistics*.

Ou Jin, Nathan N. Liu, Kai Zhao, Yong Yu, and Qiang Yang. 2011. Transferring topical knowledge from auxiliary long texts for short text clustering. In *Proceedings of the 20th ACM international conference on Information and knowledge management*.

Thomas K Landauer, Peter W. Foltz, and Darrell Laham. 1998. An introduction to latent semantic analysis. *Discourse Processes*, 25.

H. J. Lee and Sung Joo Park. 2007. Moners: A news recommender for the mobile web. *Expert Syst. Appl.*, 32(1):143–150.

Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett. 2006. Sentence similarity based on semantic nets and corpus statistics. *IEEE Transaction on Knowledge and Data Engineering*, 18.

Hao Li, Yu Chen, Heng Ji, Smaranda Muresan, and Dequan Zheng. 2012. Combining social cognitive theories with linguistic features for multi-genre sentiment analysis. In *In Proceedings of the 26th Pacific Asia Conference on Language, Information and Computation*.

Xiaohua Liu, Shaodian Zhang, Furu Wei, and Ming Zhou. 2011. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings of the 21st National Conference on Articial Intelligence*.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*.

Daniel Ramage, Susan Dumais, and Dan Liebling. 2010. Characterizing microblogs with topic models. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*.

Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Proceedings of Advances in Neural Information Processing Systems*.

Michael Speriosui, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. 2011. Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*.

Nathan Srebro and Tommi Jaakkola. 2003. Weighted low-rank approximations. In *Proceedings of the Twentieth International Conference on Machine Learning*.

Harald Steck. 2010. Training and testing of recommender systems on data missing not at random. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *HLT-NAACL*.

George Tsatsaronis, Iraklis Varlamis, and Michalis Vazirgiannis. 2010. Text relatedness based on a word thesaurus. *Journal of Articial Intelligence Research*, 37.

Andranik Tumasjan, Timm Oliver Sprenger, Philipp G. Sandner, and Isabell M. Welpe. 2010. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *ICWSM*.

Xuerui Wang and Andrew McCallum. 2006. Topics over time: a non-markov continuous-time model of topical trends. In *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*.

Rui Yan, Mirella Lapata, and Xiaoming Li. 2012. Tweet recommendation with graph co-ranking. In *Proceedings of the 24th International Conference on Computational Linguistics*.

Lei Yang, Tao Sun, Ming Zhang, and Qiaozhu Mei. 2012. We know what @you #tag: does the dual role affect hashtag adoption? In *Proceedings of the 21st international conference on World Wide Web*.