© 2016 by Jingjing Wang. All rights reserved.

LEVERAGING MULTI-DIMENSIONAL, MULTI-SOURCE KNOWLEDGE FOR USER PREFERENCE MODELING AND EVENT SUMMARIZATION IN SOCIAL MEDIA

BY

JINGJING WANG

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Doctoral Committee:

Professor Jiawei Han, Chair
Professor Chengxiang Zhai
Professor Julia Hockenmaier
Professor Qiaozhu Mei, University of Michigan

ABSTRACT

An unprecedented development of various kinds of social media platforms, such as Twitter, Facebook and Foursquare, has been witnessed in recent years. This huge amount of user generated data are multi-dimensional in nature. Some dimensions are explicitly observed such as user profiles, text of social media posts, time, and location information. Others can be implicit and need to be inferred, reflecting the inherent structures of social media data. Examples include popular topics discussed in Twitter or Facebook, or the geographical clusters based on user check-in activities from Foursquare. It is of great interest to both research communities and commercial organizations to understand such heterogeneous data and leverage available information from multiple dimensions to facilitate social media applications, such as user preference modeling and event summarization. This dissertation first presents a general discriminative learning approach for modeling multi-dimensional knowledge in a *supervised setting.* A learning protocol is established to model both explicit and implicit knowledge in a unified manner, which applies to general classification/prediction tasks. This approach accommodates heterogeneous data dimensions with a significant boosted expressiveness of existing discriminative learning approaches. It stands out with its capability to model *latent features*, for which arbitrary generative assumptions are allowed. Besides the multi-dimensional nature, social media data are unstructured, fragmented and noisy. It makes social media data mining even more challenging that a lot of real applications come with no available annotation in an *unsupervised setting.* This dissertation addresses this issue from a novel angle: external sources such as news media and knowledge bases are exploited to provide supervision. I describe a unified framework which links traditional news data to Twitter and enables effective knowledge discovery such as event detection and summarization.

To my family for all their love.

ACKNOWLEDGMENTS

This thesis would not have been possible without the support of many people.

I am profoundly grateful to my adviser, Dr. Jiawei Han, who has always been encouraging me to aim high and guided me on the road of data mining research with his extraordinary insight and exceptional expertise. I'm not only inspired by his dedication, passion, and practical approach to research, but also always touched by his kindness, patience and generous support in helping his students to achieve the best they can. I feel extremely lucky to be a member of the fabulous data mining group led by Dr. Han.

I greatly appreciate the guidance and support of my Ph.D. committee, Dr. ChengXiang Zhai, Dr. Julia Hockenmaier, and Dr. Qiaozhu Mei, and am grateful to them for their invaluable comments on my dissertation.

Many thanks to my friends and research collaborators: Dr. Changsung Kang and Dr. Yi Chang from Yahoo! Labs, Dr. Mudhakar Srivatsa and Dr. Raghu Ganti from IBM T.J. Watson Research Center, Dr. Zhenhui Li, Huan Gui, Dr. Hongbo Deng, Jialu Liu, Min Li, Wenzhu Tong, Dr. Meng Jiang, Dr. Quan Yuan, Hongkun Yu, Guangyu Zhou, Yucheng Chen, Keyang Zhang, Fangbo Tao, Haoyan Cai, Dr. Tim Hanratty, Dr. Xiuli Ma, Chi Wang, Ming Ji, Xiao Yu, Xide Lin, Quanquan Gu, Rui Li, Marina Danilevsky, Lv An Tang, Mingjie Qian, Hong Cheng, Zhijun Yin, Tim Weninger, Chenguang Wang, Bhaskar Prabhala, Rostyslav Korolov, Jianhua Yin, Brandon Norick, Ahmed El-Kishky, Stephen Macke, Hyung Sul Kim, Chao Zhang, Honglei Zhuang, Haoruo Peng, Jingbo Shang, Shi Zhi, Doris Xin, Xiang Ren, Yu Shi, George Brova, Xin Jin, Jing Gao, Lu Su, and every

DAIS group member, who made my academic life much more enjoyable by providing helpful discussions and constantly cheering me up.

A very special thanks to Jump Labs at the research park for their support during my last year at UIUC to explore a new field with my experience in data mining, and for sponsoring my Ph.D. defense.

Finally, and above all, I would like to thank my family for their love and support during this thesis. I owe my deepest gratitude to my parents for their love, understanding, and encouragement to pursue a doctoral degree at the first place, as well as my boyfriend Xiaolong for his unwavering mental, emotional, and academic support. This thesis is dedicated to them.

This thesis was supported in part by the U.S. Army Research Laboratory under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), National Science Foundation IIS-1017362, IIS-1320617, and IIS-1354329, HDTRA1-10-1-0120, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative (www.bd2k.nih.gov).

TABLE OF CONTENTS

Chapter 1

Introduction

An unprecedented development of various kinds of social media platforms, such as Twitter, Facebook and Foursquare, has been witnessed in recent years. People share their daily activities and thoughts on these platforms via check-ins, posts and comments. The ever-increasing popularity of social media and the huge amount of available user generated data create great opportunities for both research communities and commercial organizations, leading to many important, real-world applications driven by the real need. For example, modeling users' topic preference by analyzing their social media posts is one of the fundamental tasks in advertising. It helps recommender sytstems to push relevant content to users, from news articles and research findings, to movies and operas. Detecting the most popular events discussed on social media platforms and summarizing them properly also help policy makers to understand the public's opinions and meet their needs.

Yet the heterogeneous data dimensions and the extremely noisy social media utterances pose tremendous challenges on understanding and mining such data. The objective of this dissertation is to study effective approaches to model multi-dimensional social media data, in order to help users discover and explore useful knowledge. It addresses the following challenges:

- *Data Complexity.* Social media data are *multi-dimensional* in nature. There exist heterogeneous data dimensions, such as geographic coordinates, entities, time, words, topics, regions, etc. Some dimensions are explicitly revealed, such as user profiles, the text of posts, time, and location information. Others can be implicit and need to be inferred, reflecting the inherent structures of social media data. Examples include popular topics discussed in Twitter or Facebook, or the geographical clusters based on user check-in activities from Foursquare. While we have seen abundant existing research in social media data, there lacks a methodology to model both explicit and implicit knowledge in a principled manner.

- *Data quality.* Social media data are unstructured, fragmented and noisy. Useful information are often buried in huge amount of irrelevant information. Furthermore, most knowledge discovery tasks may not have readily available annotated training data, which desire unsupervised or weakly supervised models that can effectively extract knowledge out of massive noisy data.

1.1 A Principled Method for Modeling Multi-Dimensional User Preference on Social Media Platforms

The first contribution of this dissertation is a minimax entropy model for learning check-in preference on social media platforms, which is a single unified discriminative learning approach to model multidimensional knowledge in a supervised setting. This approach offers a learning protocol that applies to general classification/prediction task, which accommodates heterogeneous data dimensions with a significant boosted expressiveness of existing discriminative learning approaches. It stands out with its capability to model *latent features*, which can be carved by any parametric forms with arbitrary generative assumptions. Flexible as the way latent features are defined by parametric forms, the parameters governing the latent features are inferred jointly with the learning task in a principled way. These parameters serve to explain the inherent structure of the learning task. The minimax model is presented in the context of a concrete application: learning users' check-in preference in social media platforms. It is demonstrated to be capable of modeling user preference in an optimized manner.

Check-in preference of users is a fundamental component of Point-of-Interest (POI) prediction and recommendation in social media. It is a perfect example where multi-dimensional information jointly affect the final outcome. A user's check-in is affected at multiple dimensions, such as the particular time, popularity of the place, his/her category and geographic preference, etc. With the geographic preferences modeled as latent features and the rest as explicit features, our approach provides an in-depth understanding of users' time-varying preferences over different POIs. Meanwhile, a reasonable representation of the hidden geographic clusters based on user preference is learned in a joint manner. Experimental results based on the task of POI prediction/recommendation with real-world datasets demonstrate that our approach significantly outperforms the state-of-art models where only a subset of dimensions are considered, or different dimensions are combined in an ad-hoc manner.

1.2 Leverage External Sources for Multi-Dimensional Knowledge Discovery in Social Media

Besides the multi-dimensional nature, social media data are unstructured, fragmented and noisy. It makes social media data mining even more challenging that a lot of real applications come with no available annotation in an *unsupervised setting*. This dissertation studies the practical problem of event detection and summarization in social media. It addresses the above issues from a novel angle where external sources such as news media and knowledge bases are exploited to provide supervision. A major event usually has repercussions on both news media and social media sites such as Twitter. Unlike the "free-style" social media posts, news articles are written in formal languages, concentrated on important facts, and have a broad coverage of major events. These properties make news an ideal source for guiding knowledge discovery in social media.

I describe a unified framework which links traditional news data to Twitter and enables effective knowledge discovery. The proposed framework consists of a novel and efficient multidimensional topic model for event detection, and an effective linking module combining information retrieval and a bootstrapped dataless classification scheme. The topic model learns accurate multi-dimensional descriptors (anchors) of events from news. Then the linking module connects tweets and news via these anchors. The linking module is completely unsupervised, yet elegantly handles the challenges of selecting informative tweets under overwhelming noise and bridging the vocabulary gap between news and tweets. This framework complements the aforementioned discriminative approach to model multidimensional knowledge under a completely unsupervised setting.

In addition, I developed an online system running on near real-time data that demonstrates the effectiveness of our approach. With a given time period as the input, our system displays informative presentations of the major events with entity graphs, time spans, news summaries and tweet highlights to facilitate user digestion.

1.3 Leverage Social Media for Customized Event Profiling in Traditional News Media

Traditional news media offer high-quality reference context for social media, which helps to identify meaningful information in social media. However, the gain does not have to be one-way. In fact, social media posts reflect people's true feelings and what they really care about. Integrating social media brings a new perspective to the traditional news mining tasks, inspiring a broad spectrum of applications such as opinion-worthy event detection and twitter-customized news summarization.

I investigate customized event profiling as the third part of this dissertation. Numerous research efforts have been aimed at news event detection and summarization. Various forms of local and global textual features have been extensively exploited to advance the state-of-art methods. Leveraging social media data for traditional mining tasks, however, is much less explored. With a reliable technique to align news data and social media data, I further seek the technology to customize news event profiling with social impact. The aspects that attract people's attention (*i.e.*, popular on social media platforms) are emphasized in the event profiles.

To this end, a novel graph-based method is proposed which leverages massive tweets to customize news event profiling. A propagation model which seamlessly combines global and local context is developed on a *news-content units-tweets* tripartite graph to effectively propagate social impact information from tweets to news. The ranking of news sentences are influenced by tweets in a way that the highly ranked news sentences are more interesting to the users. Such interestingness is measured by the popularity of the tweets. The event profiles can be readily used to generate summaries for events, and they are expected to better reflect people's interest. Although our method is designed to capture the aggregate trends of the public's interest, it applies to fine grained user groups as well. Given different user groups, either by age, by gender, or by location, if we confine tweets to each group and obtain the corresponding customized profile, we will be able to tell the interest drift from one group to another. This not only can benefit real-world applications such as personalized news recommendation, but also can be of great interest to social scientists.

Organization of the Dissertation The first chapter introduces the challenges and problems studied in this dissertation. Chapter 2 presents the the findings and methodologies for modeling multi-dimensional user preference on social media platforms. Chapter 3, 4 and 5 further introduce external sources to the multi-dimensional setting, in the context of integrated studies of new media and social media. Chapter 6 concludes the dissertation.

Chapter 2

A Minimax Entropy Approach to Modeling Multi-Dimensional User Preference on Social Media Platforms

2.1 Overview

Modeling the time aware check-in preference of users is a perfect example where multi-dimensional information jointly affect the final outcome. It is also the fundamental component of location prediction and location recommendation on social media platforms. As the check-in feature becomes increasingly popular in major social media platforms such as Foursquare, Facebook, etc., numerous research efforts have been aimed at mining users' check-in behaviors. In this Chapter, we consider the problem of modeling users' time-aware check-in preferences. Formally, our goal is to learn a time-aware distribution over POIs for each user: $p(l|u,t)$, where u denotes a user, t denotes a time point, l denotes a POI and $p(l|u,t)$ denotes the conditional probability that l is checked in given that the user is u and the time point is t. This distribution allows us to predict what are the top places a user would like to check in at a given time, which can be of great interest to both business owners and advertisement providers.

A discriminative learning framework is proposed where a subset of the features are allowed to be latent. In contrast to the standard discriminative learning protocol (e.g. SVM, logistic regression) where features are readily available before training, we introduce the concept of *latent features*. The value of a latent feature is not known before training, but is specified by a parametric form with unknown parameters. The parametric form can capture arbitrary underlying assumptions to describe the feature. For example, if a set of latent features are cluster indicators, the parameters can specify the underlying clustering structure such as a

Gaussian mixture membership model. During the training process, the latent parameters are jointly inferred with the classification task. We illustrate in the following paragraphs why this is the desired strategy.

Why maximum entropy?

A naive way to estimate $p(l|u,t)$ is simple counting. For each user u at time t, we can get the histogram of POIs (l's) and view it as the objective distribution. While this distribution perfectly fits the seen data, it is not generalizable, *i.e.*, it can never predict unvisited POIs for users and will fail to generate outputs for unseen time points.

We prefer a model which *explains the seen data well and meanwhile has good generalizability.* To this end, instead of exactly matching $p(l|u,t)$ to the empirical distribution, it is natural to extract features from the `user-time-POI` $\langle utl \rangle$ tuples and impose the constraints that $p(l|u,t)$ match the empirical statistics in the feature space. Among these qualified distributions, we select the distribution with the maximum entropy as the optimal distribution, as it assumes least bias on the model beyond the constraints we specify [26].

Why minimax entropy? (Why latent features? Why should they be jointly learned?)

User preferences over POIs can be affected by explicit features such as the category of a POI, the day of a week, etc., meanwhile it can also be affected by the more ambiguous features such as the geographic region, which is less clear how to encode as features effectively. For example, it is not straightforward to draw the boundary for "downtown Manhattan" or to classify if a POI belongs to it. Therefore, we introduce latent features to model this kind of ambiguity. Taking the geographic feature as an example, we can assume there exist geographic clusters, each of which is specified by latent parameters: a center (coordinates of latitude and longitude) and a radius (a positive real number). Given a POI, we define a weight vector over different clusters as a *latent* feature vector, where the weight on each cluster is determined by a parametric function which takes the latitude-longitude of the POI as input. With both explicit and latent features, we propose a minimax entropy approach to jointly learn the latent parameters together with the check-in preferences ($p(l|u,t)$). The joint learning approach is motivated by the fact that *the clustering structure is not only determined by geographic proximity, but also affected by how well it explains user check-ins.* For example, even if two POIs are very close to each other geographically, if they have never been visited by the same user, it may not be appropriate to put them into the same cluster. In sum, the jointly learned geographic clusters are specially tailored to boost the learning task's performance rather than just provide a standalone clustering results.

Contributions

- We propose a single unified minimax entropy approach which elegantly leverages explicit features and latent features for user preference modeling. It boosts the flexibility and expressiveness of the standard discriminative learning models significantly.

- Flexible as the way latent features are defined by parametric forms, the parameters governing the latent features are recovered jointly with the learning task in a principled way, which serve to explain the inherent structure of the learning task.

- We demonstrate the effectiveness of our approach in the context of check-in preference learning with its rich types of information. It opens up a promising direction for preference learning with multidimensional heterogeneous knowledge.

The rest of this chapter is organized as follows. Section 2.2 details the modeling of users, POIs and the way we specify the geographic clusters; and then formally defines the problem. We introduce our framework for check-in preference modeling in Section 2.3, review related work in Section 2.4, report our experimental results on real-world data in Section 2.5, and summarize this study in Section 2.6.

Table 2.1: Summary of Notations

Symbol	**Description**
u, U	a user, user set
t, T	a time index, time index set; $day(t)$ and $hour(t)$ denote the day index and hour index of t, respectively
l, L	a POI, POI set; $cat(l)$ denotes the category of l
C	category set
$\mathbf{o} = (o_1, o_2, ..., o_R)$	the centers of the geographic clusters
$\mathbf{r} = (r_1, r_2, ..., r_R)$	the radiuses of the geographic clusters
$\mathbf{c}^u = (c^u_1, c^u_2, ..., c^u_C)$	u's category preference
$\mathbf{g}^u = (g^u_1, g^u_2, ..., g^u_R)$	u's geographic preference
$\mathbf{c}^l$	l's one-hot encoding of its category
$\mathbf{g}^l = (g^l_1, g^l_2, ..., g^l_R)$	l's weights on different regions
p^l	l's global popularity
$\mathbf{d}^l = (d^l_1, d^l_2, ..., d^l_7)$	l's daily popularity profile
$\mathbf{h}^l = (h^l_1, h^l_2, ..., d^l_{24})$	l's hourly popularity profile
$\pi_{utl} = p(l\|u, t)$	the conditional probability of checking in at POI l given a user u and time t
$\tilde{\pi}_{ut} = \tilde{p}(u, t), \tilde{\pi}_{utl} = \tilde{p}(l\|u, t)$	the empirical distributions estimated from data
Π	the true check-in preference distribution

2.2 Problem Formulation

In this section, we define the POI profiles and user profiles with both explicit knowledge and the latent geographic clustering structure governed by latent parameters. Then we give the formal definition of check-in preference modeling. The notations used in this chapter are summarized in Table 5.1.

Let U, T, L, C be the user set, time set, POI set and category set respectively. Our data contains the histories of user check-ins.

DEFINITION 1 (Check-in)**.** *A check-in is denoted by a user-time-POI tuple* $\langle utl \rangle$*, where* $u \in U, t \in T$ *and* $l \in L$*. Each POI* l *is associated with its category, latitude and longitude. The time is represented by the day of week and hour of day*[1]*.*

[1] There are 7x24 unique values in T under this setting. However, one can index time with finer or coarser granularity as well. Overlapped time intervals are also allowed.

DEFINITION 2 (Region)**.** *A region is a geographic cluster defined by the latitude and longitude of the center* $o = (o_{lat}, o_{lon})$ *and a radius* $r > 0$*. The* (o, r)*'s are the latent parameters.*

DEFINITION 3 (POI Profile)**.** *A POI* l *is represented by a profile*[2] $\rho(l) = [\mathbf{c}^l, \mathbf{g}^l(\mathbf{o}, \mathbf{r}), \mathbf{d}^l, \mathbf{h}^l, p^l]$.

- $\mathbf{c}^l$ (a one-hot encoding of l's category): $\mathbf{c}^l$ has $c_i^l = 1$ if the i-th category in C is the category of l and 0 otherwise.

- $\mathbf{g}^l$ (the geographic profile of l): The geographic profile of a POI is modeled by a weight vector over different regions. The weight is determined by the POI's distance to the center of a region and the radius of the region:

  $$g_i^l = \exp(-\frac{dist(l, o_i)}{r_i}) \tag{2.1}$$

  where $dist(\cdot, \cdot)$ is the Euclidean distance[3].

  When $dist(l, o) = 0$, the weight reaches its maximum 1; as $dist(l, o)$ becomes larger, the weight decreases towards 0. The radius r controls the decreasing speed w.r.t $dist(l, o)$. A smaller r indicates a more concentrated cluster, *i.e.*, the weight decreases drastically as the distance increases. Note that the weight function does not necessarily have to be defined in this way. A function that can satisfy the desired properties suffices.

- p^l (global popularity of l): The global popularity of a POI is defined as the total number of check-ins at this POI.

- $\mathbf{d}^l, \mathbf{h}^l$ (the daily popularity profile and hourly popularity profile of l): POIs have time varying popularity as well. For example, a nightclub has its rush hours at night but is either closed or rarely visited before sunset. We compute the time varying popularity

[2] We use bold letters to denote column vectors. The comma between column vectors indicates a vertical stack of the vectors.

[3] Other distance measures apply as well.

based on the aggregate statistics from all users. d_i^l is the proportion of check-ins at l that happen on the ith day of a week and h_i^l is the proportion of check-ins at l that happen on the ith hour of a day.

DEFINITION 4 (User Profile). *A user u is represented by a profile $\rho(u) = [\mathbf{c}^u, \mathbf{g}^u(\mathbf{o}, \mathbf{r})]$.*

- $\mathbf{c}^u$ (user u's preference over categories): We define user u's preference of category i (*i.e.*, c_i^u) to be the proportion of his/her check-ins that fall into category i.
- $\mathbf{g}^u$ (user u's preference over regions): In addition to the category preference, users are also characterized by their geographic preferences over different regions. We define user u's geographic preference of a region i (*i.e.*, g_i^u) to be the aggregate weights at region i of all his/her check-ins .

We are now able to formulate the check-in preferences modeling problem as follows.

PROBLEM 1 (Check-in Preferences Modeling). *Given a training set of user check-in tuples, where each tuple $\langle utl \rangle$ is associated with a user profile $\rho(u)$ and a POI profile $\rho(l)$, jointly learn the conditional probability of checking in at POI l given a user u and time t, denoted by $\pi_{utl} = p(l|u,t), \forall u, t, l$; and the geographic clustering structure governed by latent parameters $\mathbf{o}$ and $\mathbf{r}$.*

2.3 A Minimax Entropy Approach for Modeling Check-in Preferences

In this section, we first assume the latent parameters are given, *i.e.*, all the features are explicit, and present the maximum entropy (MaxEnt) model for learning the check-in preferences. Then we present the proposed minimax entropy model which estimates the latent parameters jointly with the preference learning.

2.3.1 A Maximum Entropy Model

The most aggressive way to model the check-in preferences is just to let π_{utl} equal the empirical distribution[4] $\tilde{\pi}_{utl} = \frac{\#\langle utl \rangle}{\sum_l \#\langle utl \rangle}$. However, this will overfit the data and is not generalizable. We want to construct a model which explains the seen data well, and meanwhile has good generalizability. To this end, we adopt the maximum entropy principle to specify $\{\pi_{utl}\}$, *i.e.*, we choose the most "uniform" distribution with carefully chosen constraints instantiated by features. These constraints should guarantee that our model accords with the data statistics we feel essential in modeling the check-in preferences.

Features Based on Multidimensional Preferences

We consider the following factors to model check-in preferences: temporal preference, category preference, geographic preference and the popularity of the POI. Consider the following scenario: on a Friday evening, Alice just finished yet another week of hard work; she would like to have a great dinner at a seafood restaurant and then she figures a popular Boiling Crab branch is just nearby. Then it is very likely she checks in at this place. We design the following features to instantiate the constraints which will be used to specify our model $\{\pi_{utl}\}$.

- **Category Preference**. The extent to which a POI l matches a user u's category preference is estimated by $f_c(\langle utl \rangle) = \mathbf{c}^{u^T}\mathbf{c}^l$.
- **Geographic Preference**. The extent to which a POI l matches a user u's geographic preference is estimated by $f_g(\langle utl \rangle) = \mathbf{g}^{u^T}\mathbf{g}^l$.
- **Temporal Preference**. If we represent each time index t with two one-hot encodings: $\mathbf{d}^t$, $\mathbf{h}^t$ for the day and hour respectively, the extent to which a POI l's daily

[4] In this chapter, we use $\#\langle utl \rangle$ to denote the number of appearances of the check-in tuple $\langle utl \rangle$ in the data, and $\#$ to denote the total number of check-ins. We use " $\tilde{}$ " to denote the empirical distribution. Later we will also see $\tilde{p}(u,t) = \tilde{\pi}_{ut} = \frac{\sum_l \#\langle utl \rangle}{\#}$

popularity matches a time t is estimated by $f_d(\langle utl\rangle) = \mathbf{d}^{l^T}\mathbf{d}^t$, and hourly popularity by $f_h(\langle utl\rangle) = \mathbf{h}^{l^T}\mathbf{h}^t$.

- **Popularity Preference**. As more popular POIs usually would expect more check-ins, we assign a popularity preference for each POI without distinguishing users. $f_p(\langle utl\rangle) = p^l$.

Let $\mathbf{f} = [f_c, f_g, f_d, f_h, f_p]^T$. It[5] measures how a POI matches a user's preference at a particular time. We employ constraints that require our model to accord with the data at each dimension of the preferences, *i.e.*, the model distribution matches the empirical distribution at the feature space:

$$\begin{aligned} &\mathbb{E}_{\pi}(\mathbf{f}) = \mathbb{E}_{\tilde{\pi}}(\mathbf{f}) \\ i.e., \quad &\sum_{u,t,l} \tilde{p}(u,t)p(l|u,t)\mathbf{f} = \sum_{u,t,l} \tilde{p}(u,t)\tilde{p}(l|u,t)\mathbf{f} \\ i.e., \quad &\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f} \end{aligned}$$

where $\mathbb{E}$ denotes expectation. Note that we do not model the joint distribution of u and t (*i.e.*, $p(u,t)$) since the goal is to predict l given u and t. We let $p(u,t) = \tilde{p}(u,t) = \tilde{\pi}_{u,t}$. The model parameters[6] here contain $\{\pi_{utl}, \forall u, t, l\}$ only. This also classifies our problem as a discriminative learning task (as opposed to generative learning).

A Maximum Entropy Model with Fixed Latent Parameters

With the constraints defined above, we formulate our MaxEnt model in this section.

[5] A complete notation should be $\mathbf{f}(\langle utl\rangle) = (f_c(\langle utl\rangle), f_g(\langle utl\rangle), f_d(\langle utl\rangle), f_h(\langle utl\rangle), f_p(\langle utl\rangle))^T$, in the following of this chapter, we omit $(\langle utl\rangle)$ for brevity and readability.

[6] We slightly abuse the terminology *parameter*. Model parameters refer to π_{utl} and latent parameters refer to $(\mathbf{o}, \mathbf{r})$.

The conditional entropy of π_{utl} is given by

$$H(\pi) = -\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} = -\mathbb{E}_{\pi}(\ln \pi_{utl})$$

As discussed in the previous section, we constrain the distribution π to a set $\mathcal{C}$ of allowed probability distributions:

$$\mathcal{C} = \{\pi | \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f}\}$$

By the MaxEnt principle, we should select a model from $\mathcal{C}$ with maximum $H(\pi)$:

$$\pi^* = \arg\max_{\pi \in \mathcal{C}} H(\pi)$$

Therefore we have the following MaxEnt model for π:

$$\max_{\pi} -\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} \tag{2.2}$$

$$s.t. \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f} \tag{2.3}$$

$$\sum_{l} \pi_{utl} = 1 \quad \forall u, t \tag{2.4}$$

$$\pi_{utl} > 0 \quad \forall u, t, l \tag{2.5}$$

Note that equation (2.3) is a vector form of $|\mathbf{f}| = 5$ constraints, corresponding to the 5 dimensional preferences.

We solve the constrained optimization problem in the dual space:

Primal Dual Conversion. The Lagrangian of the MaxEnt problem is

$$\begin{aligned}\mathcal{L} = &-\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} \\ &+ \sum_{\alpha} w_\alpha (\sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} f_\alpha - \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl} f_\alpha) \\ &+ \sum_{u,t} \eta_{u,t} (\sum_{l} \pi_{utl} - 1)\end{aligned}$$

where $\{w_\alpha\}$ and $\{\eta_{u,t}\}$ are the Lagrange multipliers.

Let $\dfrac{\partial \mathcal{L}}{\partial \pi_{utl}} = 0$, we have

$$\begin{aligned}&- \tilde{\pi}_{ut}(1 + \ln \pi_{utl}) + \sum_{\alpha} w_\alpha (\tilde{\pi}_{ut} f_\alpha) + \eta_{u,t} = 0 \\ &\iff \ln \pi_{utl} = \sum_{\alpha} w_\alpha f_\alpha + \frac{\eta_{u,t}}{\tilde{\pi}_{ut}} - 1\end{aligned}$$

Apply the constraint $\sum_l \pi_{utl} = 1 \quad \forall u, t$, we can get

$$\pi_{utl} = \frac{\exp(\sum_\alpha w_\alpha f_\alpha)}{\sum_l \exp(\sum_\alpha w_\alpha f_\alpha)} \forall u, t, l \tag{2.6}$$

Plugging Equation (2.6) into $\mathcal{L}$ gives that $\mathcal{L}$ is the minus log likelihood of the data. Maximizing the primal problem becomes minimizing the dual problem, which turns out to be maximizing the log likelihood of the data with π_{utl} specified by Equation (2.6). Therefore $\mathbf{w}^*$ is the maximum likelihood estimation:

$$LL = \sum_{utl} \tilde{\pi}_{ut}\tilde{\pi}_{utl} \ln \pi_{utl}, \quad \mathbf{w}^* = \arg\min_{\mathbf{w}} -LL$$

where π_{utl} is of the form given in Equation (2.6).

□

In sum, we have the following form of π_{utl}:

$$\pi_{utl} = \frac{\exp(\mathbf{w}^T \mathbf{f}(\langle utl \rangle))}{\sum_l \exp(\mathbf{w}^T \mathbf{f}(\langle utl \rangle))} \quad \forall u, t, l \tag{2.7}$$

where $\mathbf{w}$ is the Lagrange coefficients. Solving the primal problem turns out to be maximizing the log likelihood of the data with π_{utl} specified by Equation (2.7). And we obtain the optimal $\mathbf{w}^*$ from the maximum likelihood estimation:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} LL \tag{2.8}$$

$$LL = \sum_{utl} \tilde{\pi}_{ut} \tilde{\pi}_{utl} \ln \pi_{utl} \tag{2.9}$$

Finally, the solution for the primal problem is given by:

$$\pi^*_{utl} = \frac{\exp(\mathbf{w}^{*T} \mathbf{f})}{Z_{ut}}, \quad Z_{ut} = \sum_l \exp(\mathbf{w}^{*T} \mathbf{f}) \quad \forall u, t, l$$

where $\mathbf{w}^*$ is the optimal Lagrange coefficients, each element of which corresponds to a constraint in Equation (2.3).

2.3.2 Recovering Latent Parameters via Minimax Entropy

In the previous section, we have completed the discussion for the case where we assume the latent parameters are given so that all the features are explicit. Now let us bring the latent features back. We have f_g as a *latent feature* which is parameterized by $(\mathbf{o}, \mathbf{r})$. Therefore $\mathbf{w}^*$ is also parameterized by $(\mathbf{o}, \mathbf{r})$. The optimal solution is thus $\pi^*(\mathbf{o}, \mathbf{r})$:

$$\pi^*_{utl}(\mathbf{o}, \mathbf{r}) = \frac{\exp(\mathbf{w}^*(\mathbf{o}, \mathbf{r})^T \mathbf{f}(\langle utl \rangle)(\mathbf{o}, \mathbf{r}))}{\sum_l \exp(\mathbf{w}^*(\mathbf{o}, \mathbf{r})^T \mathbf{f}(\langle utl \rangle)(\mathbf{o}, \mathbf{r}))} \quad \forall u, t, l$$

We propose that ***the optimal*** $(\mathbf{o}, \mathbf{r})$ ***should be chosen such that the maximized conditional entropy*** $H(\pi^*(\mathbf{o}, \mathbf{r}))$ ***is minimized*** and justify this statement in this section.

To measure the quality of the check-in preference distribution, we use the standard Kullback-Leibler (KL) divergence [29] from $\pi^*(\mathbf{o}, \mathbf{r})$ to the true user check-in preference Π. Π is the true conditional distribution: $\Pi_{utl} = p_{true}(l|u,t)$[7]. The optimal $(\mathbf{o}, \mathbf{r})$ should give the smallest KL divergence:

$$(\mathbf{o}^*, \mathbf{r}^*) = \underset{\mathbf{r}>0,\mathbf{o}}{\arg\min}\, KL(\Pi, \pi^*(\mathbf{o}, \mathbf{r}))$$

where

$$\begin{aligned} KL(\Pi, \pi^*(\mathbf{o}, \mathbf{r})) &= \mathbb{E}_{\Pi}(\ln \Pi_{utl}) - \mathbb{E}_{\Pi}(\ln \pi^*_{utl}) \\ &= -\mathbb{E}_{\Pi}(\ln \pi^*_{utl}) - H(\Pi) \end{aligned}$$

The difficulty here is that the true distribution Π is unknown, thus we cannot directly evaluate the first term. However, under the assumption that our sample size is reasonably large, which means the expected feature statistics $\mathbb{E}_{\Pi}(\mathbf{f})$ can be approximated exactly by neglecting the estimation errors in the observed statistics $\mathbb{E}_{\tilde{\pi}}(\mathbf{f})$, we obtain the following theorem.

THEOREM 1. *The KL divergence from* $\pi^*(\mathbf{o}, \mathbf{r})$[8] *to the true distribution* Π *is given by* $KL(\Pi, \pi^*) = H(\pi^*) - H(\Pi)$

Proof. We need to prove $\mathbb{E}_{\Pi}(\ln \pi^*_{utl}) = -H(\pi^*)$. As shown before, π^* has the following form:

$$\pi^*_{utl} = \frac{\exp(\mathbf{w}^{*T}\mathbf{f})}{Z_{ut}}, Z_{ut} = \sum_{l} \exp(\mathbf{w}^{*T}\mathbf{f}) \quad \forall u, t, l$$

[7] As before, we do not model $\Pi_{ut} = p_{true}(u,t)$ and let $\Pi_{ut} = \tilde{\pi}_{ut}$.

[8] For brevity, we use π^* short for $\pi^*(\mathbf{o}, \mathbf{r})$ in this proof

where $\mathbf{w}^*$ is the optimal Lagrange coefficients. Hence we have

$$\begin{aligned}
\mathbb{E}_{\Pi}(\ln \pi^*_{utl}) &= \mathbb{E}_{\Pi}(\mathbf{w}^{*T}\mathbf{f}) - \mathbb{E}_{\Pi}(\ln Z_{ut}) \\
&= \mathbb{E}_{\tilde{\pi}}(\mathbf{w}^{*T}\mathbf{f}) - \mathbb{E}_{\tilde{\pi}}(\ln Z_{ut}) \\
&\qquad\qquad \text{by } \mathbb{E}_{\tilde{\pi}}(\mathbf{f}) = \mathbb{E}_{\Pi}(\mathbf{f}) \\
&= \mathbb{E}_{\pi^*}(\mathbf{w}^{*T}\mathbf{f}) - \mathbb{E}_{\pi^*}(\ln Z_{ut}) \\
&\qquad\qquad \text{by Equation (2.3)} \\
&= \mathbb{E}_{\pi^*}(\ln \pi^*_{utl}) = -H(\pi^*)
\end{aligned}$$

and the result follows. □

As the entropy of Π is fixed, and the entropy of π^* is parameterized by $(\mathbf{o}, \mathbf{r})$, in order to minimize $KL(\Pi, \pi^*(\mathbf{o}, \mathbf{r}))$, we conclude that the latent variables should be estimated by minimizing the maximized entropy:

$$(\mathbf{o}^*, \mathbf{r}^*) = \arg\min_{\mathbf{o},\mathbf{r}} \sum_{u,t,l} -\tilde{\pi}_{ut}\pi^*_{utl}(\mathbf{o}, \mathbf{r}) \ln \pi^*_{utl}(\mathbf{o}, \mathbf{r}) \tag{2.10}$$

Therefore, we obtain our entire minimax entropy framework as summarized in the following program:

$$\begin{aligned}
&\min_{\mathbf{o},\mathbf{r}} \max_{\pi} - \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} \ln \pi_{utl} && (2.11) \\
&s.t. \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl}\mathbf{f} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl}\mathbf{f} && (2.12) \\
&\sum_{l} \pi_{utl} = 1 \quad \forall u, t && (2.13) \\
&\pi_{utl} > 0 \quad \forall u, t, l && (2.14) \\
&r_i > 0 \quad \forall i && (2.15)
\end{aligned}$$

2.3.3 The Learning Algorithm

Algorithm 1: The learning Algorithm for the Minimax Entropy Approach of Check-in Preferences Modeling

```
Input: A user check-in database $\{\langle utl \rangle\}$
Output: Check-in preference $\{\pi_{utl}\}, \forall u, t, l$; geographic clustering parameters $(\mathbf{o}, \mathbf{r})$
Do a K-means clustering on the latitude-longitude coordinates of the POIs. Initialize $\mathbf{o}^*$ and $\mathbf{r}^*$ to be centers and average distances to the centers.
for iter = 1:Maxiter do
    MaxEnt step. With $(\mathbf{o}, \mathbf{r})$ fixed to $(\mathbf{o}^*, \mathbf{r}^*)$, solve the MaxEnt problem to obtain $\mathbf{w}^*$.
        $\mathbf{w}^* = \arg\max_{\mathbf{w}} LL$
        where $LL(\mathbf{w}, \mathbf{o}^*, \mathbf{r}^*) = \sum_{utl} \tilde{\pi}_{ut} \tilde{\pi}_{utl} \ln \pi_{utl}$,
        $\pi_{utl} = \frac{\exp(\mathbf{w}^T \mathbf{f}(\mathbf{o}^*, \mathbf{r}^*))}{\sum_l \exp(\mathbf{w}^T \mathbf{f}(\mathbf{o}^*, \mathbf{r}^*))}$
    MinEnt step. With $\mathbf{w}$ fixed to $\mathbf{w}^*$, estimate the latent parameters $(\mathbf{o}, \mathbf{r})$.
        $(\mathbf{o}^*, \mathbf{r}^*) = \arg\min_{\mathbf{r}>0, \mathbf{o}} -LL$
        where $LL(\mathbf{w}^*, \mathbf{o}, \mathbf{r}) = \sum_{utl} \tilde{\pi}_{ut} \tilde{\pi}_{utl} \ln \pi_{utl}$,
        $\pi_{utl} = \frac{\exp(\mathbf{w}^{*T} \mathbf{f}(\mathbf{o}, \mathbf{r}))}{\sum_l \exp(\mathbf{w}^{*T} \mathbf{f}(\mathbf{o}, \mathbf{r}))}$
end
```

While it is hard to obtain a close form solution, we propose a neat coordinate descent learning procedure to solve the optimization problem.

With the inherent MaxEnt part converted to the dual space (see Eq. 2.8) which reduces the problem to the following form:

$$\min_{\mathbf{r}>0, \mathbf{o}} \min_{\mathbf{w}} -LL \quad i.e., \quad \min_{\mathbf{r}>0, \mathbf{o}, \mathbf{w}} -LL \tag{2.16}$$

where LL is given by Equation (2.9).

The objective now is to find the set of $(\mathbf{w}, \mathbf{o}, \mathbf{r})$ which minimizes the minus log likelihood LL of the data. This is divided to solving a MaxEnt problem (finding $\mathbf{w}^*$ with $(\mathbf{o}, \mathbf{r})$ fixed) and a MinEnt problem (finding $(\mathbf{o}^*, \mathbf{r}^*)$ with $\mathbf{w}$ fixed).

Algorithm 1 sketches the learning algorithm. First, the geographic centers are initialized by a K-means clustering; the radius for each cluster is initialized by the average distance to the center. After initialization, we solve the MaxEnt and MinEnt problems alternately to get the optimal $(\mathbf{w}, \mathbf{o}, \mathbf{r})$. Both sides of optimization are solved by the L-BFGS [35] algorithm.

Gradients for L-BFGS Updates. We derive the gradients required by L-BFGS for both MaxEnt and MinEnt steps as follows.

- The MaxEnt problem is an unconstrained optimization problem in the dual space. The gradient w.r.t $\mathbf{w}$ is given by

  $$\frac{\partial LL}{\partial w_\alpha} = \sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl} f_\alpha - \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} f_\alpha$$

  where π_{utl} is given by Equation (2.6). This is the difference between the expectations of the feature f_α from the model and the empirical mean.

  The Hessian matrix is given by

  $$\begin{aligned}\frac{\partial^2 LL}{\partial w_\alpha \partial w_\beta} =& \mathbb{E}_\pi[(\sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl} f_\alpha - \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} f_\alpha) \\ & (\sum_{u,t,l} \tilde{\pi}_{ut}\tilde{\pi}_{utl} f_\beta - \sum_{u,t,l} \tilde{\pi}_{ut}\pi_{utl} f_\beta)]\end{aligned}$$

  which is the covariance matrix of the features, and is thus positive definite[9]. This indicates that the MaxEnt problem is strictly convex and has a unique solution.

- The optimization over the latent parameters may or may not be convex, depending on the form of the chosen geographic weight function. In this paper, the problem is not

[9] only in rare cases it may be positive semi-definite

convex and L-BFGS will converge to the local minimum. We take several trials of the iteration process to approach the global minimum.

The gradient w.r.t $(\mathbf{o}, \mathbf{r})$ is given by

$$\frac{\partial LL}{\partial z_i} = \sum_{u,t,l} \tilde{\pi}_{ut} \tilde{\pi}_{utl} w_g \frac{\partial f_g}{\partial z_i} - \sum_{u,t,l} \tilde{\pi}_{ut} \pi_{utl} w_g \frac{\partial f_g}{\partial z_i}$$

where z_i can be o_{ilat}, o_{ilon} or r_i, w_g is the weight corresponding to the geographic feature f_g and

$$\frac{\partial f_g}{\partial z_i} = g_i^u \frac{\partial g_i^l}{\partial z_i} + \frac{\partial g_i^u}{\partial z_i} g_i^l$$

with g_i^l given by Equation (2.1).

□

2.4 Related Work and Discussions

Modeling the time aware check-in preference of users is the fundamental component of location[10] prediction and location recommendation. First, we review previous study on location prediction/recommendation tasks. Then we review the background of related discriminative models. In addition, connections from our approach to several standard approaches are given. We also explain how cold start issue is naturally handled by our approach.

2.4.1 Location Prediction/Recommendation

There has been a substantial amount of research on location prediction/recommendation ever since the GPS devices became widely available. The prediction and recommendation

[10] In this chapter, we use "location" and "POI" interchangeably as long as there is no ambiguity.

tasks are closely related since they both predict a list of locations which are evaluated by the prediction accuracy. There are several subtle differences though. Location prediction usually focuses more on the places which have been already visited by a user and largely depends on the time point. Therefore, spatio-temporal regularity usually plays an important role in the task. On the contrary, location recommendation task focuses more on the unvisited locations based on collaborative filtering. The recommendation may or may not be time aware as well. Unlike movie recommendations where one may not want to watch a movie he/she has already watched before, a location can be checked in repeatedly by a user. Therefore it is desirable to include the places which have been visited before in the recommendation. In this chapter, we do not distinguish between visited locations and new locations but output a distribution over all locations, where the most probable ones can be used for both prediction and recommendation.

One line of research [67, 68, 69, 70] focus on the study of GPS trajectories collected from human movements. Location prediction/recommendation on the trajectory data is a simpler task compared to the check-in data since trajectories contain consecutive movements of users which are very dense. The Nokia Research Center collected GPS data from 200 smartphone volunteers in the course of 1 year and launched a next place prediction challenge[11] in 2012. The best entries achieved prediction accuracies of above 50%.

However, location prediction/recommendation with the check-in data from LBSN is much more challenging due to the sparseness. Cheng et al. [10] propose a mixed hidden Markov model to predict the category of a user's next move and then predict the location given the category. However, while human movements may be Markovian, people usually do not check in at every POI they visit. Gao et al. [18, 19] explore the Hierarchical Pitman-Yor process [50] and view the check-in sequences as a language model to encode the historical effects. This method works much better for GPS trajectories [19] than check-in data [18]

[11] https://dl.dropboxusercontent.com/u/19156538/nokia/MDC%202012%20-%20Best%20challenge%20entries%20_%20Nokia%20Research%20Center.html

because the model also assumes dependencies between consecutive check-ins. Cho et al. [11], Gao et al. [20, 21], Yuan et al. [64] highlight the daily periodicity of check-ins and show that temporal effects have significant influence on capturing users' check-in behaviors. Gao et al. [18] incorporate geographic influence to a collaborative filtering model by assuming a power-law distribution of the pairwise check-in distances. Cheng et al. [9] extend this work to multi-center geographic distributions and combine it with a matrix factorization model. Kurashima et al. [30], Liu and Xiong [33], Liu et al. [34], Yin et al. [62] propose generative models which introduce the concept of user/location profiles. Our approach is able to incorporate the various factors from the previous work and model them in a unified way.

To make our model concrete, we defined every detail of how the features are generated. Nevertheless, the features do not necessarily have to be defined as we did in the previous sections. We have followed a natural thought that the category, geographic, temporal and popularity preferences are influential factors for a check-in. However, we can model other types of information into our learning framework as well in the forms of both explicit and latent features. For example, if the description and reviews of POIs are available, we can incorporate text features as explicit features. If social network information is available, we can incorporate friend clusters as latent features. With both explicit and latent features, our approach models ambiguous knowledge together with explicit knowledge in a unified manner to find the best possible way to utilize them.

2.4.2 Related Discriminative Models

The maximum entropy principle was first proposed byJaynes [26] in 1957. It provides a very general rationale why we should select the model with the maximum entropy. It has seen widescale applications to real world problems recently especially within the natural language processing field [5].

To the best of our knowledge, the minimax entropy principle was first proposed in the computer vision community by Zhu et al. [73], which offers a general theory and methodology for building statistical models for images (or signals) in a variety of applications. This principle consists of two parts in the original context. The first is the maximum entropy principle for feature binding (or fusion): for a given set of observed feature statistics, a distribution can be built to bind these feature statistics together by maximizing the entropy over all distributions that reproduce them. The second part is the minimum entropy principle for *feature selection*: among all plausible sets of feature statistics, the set whose maximum entropy distribution has the minimum entropy is chosen. The minimax entropy principle was applied to texture modeling, and encouraging results were obtained in experiments on a variety of texture images.

Lately Zhou et al. [71] adopted this methodology to solve a crowdsourcing problem: infer true labels out of crowdsourced noisy labels. While the maximum entropy part remains the classic formulation, the minimum entropy part aims to improve the quality of crowdsourced noisy labels. Latent *binary* variables are assigned to the true labels and inferred according to the minimax entropy principle. Substantial performance improvements were observed over existing methods.

Our proposed method takes a further step towards this direction. We prove that the minimax entropy principle still holds when the latent variables are generalized to arbitrary *parametric forms*. We provide a clean and general formulation for general classification/prediction tasks with the conventional concepts "features" and "labels", as commonly used in the standard discriminative learning methods. The features consist of explicit ones which are directly observed/computed, and latent ones which are defined by parametric forms. The latent parameters are learned in the minimum entropy part.

Previous research [41, 57, 63, 72] have also introduced hidden variables to other discriminative models such as SVMs and CRFs. However, a maximum entropy framework is able to encode meaningful semantics with its intuitive constraints. Further allowing features to be governed by parametric forms accommodates various generative assumptions. This makes our approach especially suited to user preference modeling where heterogeneous data dimensions need to be modeled.

2.4.3 Connection to Maximum Likelihood Estimation

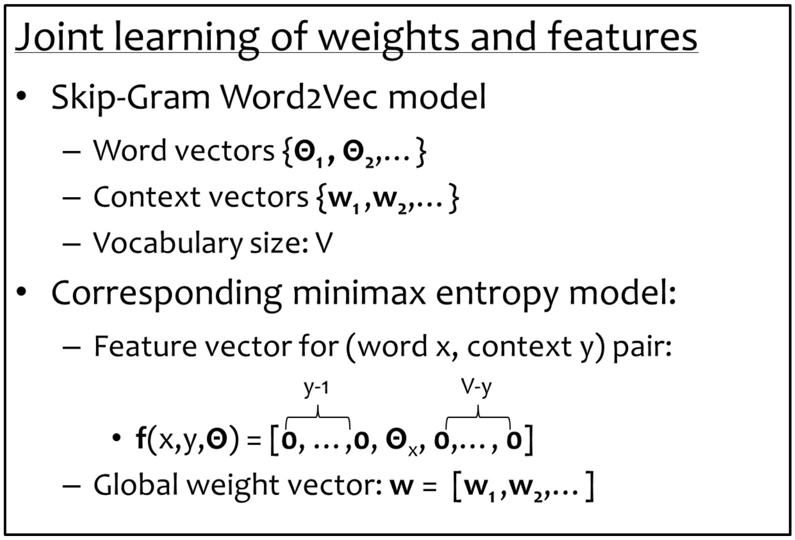
Our model has the intuitive interpretation of a discriminative maximum likelihood estimation (MLE). We have already seen that the final objective is to seek a maximum likelihood estimator $(\mathbf{w}^*, \mathbf{o}^*, \mathbf{r}^*)$ for the objective function LL, with the conditional probability defined as $p(l|u,t) = \pi_{utl} = \frac{\exp(\mathbf{w}^T\mathbf{f})}{\sum_l \exp(\mathbf{w}^T\mathbf{f})}$.

2.4.4 Connection to Matrix Factorization based Collaborative Filtering

Our model is a linear model in the sense that the prediction score is determined by $\mathbf{w}^T\mathbf{f}(\langle utl\rangle)$, and $\mathbf{w}$ is determined not only by the user check-in data, but also on the features $\mathbf{f}(\langle utl\rangle)$. In the standard matrix factorization (MF) model for recommendation where the access to meaningful information such as category, latitude and longitude is limited, it is still possible to perform prediction via purely utilizing the factorization of the user-item rating matrix R as approximated by the product of two low-rank matrices. Specifically, by carefully selecting a reasonable dimension parameter K which is much smaller than the number of users M and items N, MF approximates $R \approx U^TV$ where $U^{M\times K}$ is a user matrix and $V^{K\times N}$ is an item matrix. An interesting analogy is that the columns of U (or V) can be viewed as profiles of users (or items). However, unlike in our model where parameter estimation ($\mathbf{w}$) is performed and latent geographic clusters are learned jointly, this approach computes the prediction score in a rather simplified manner as u^Tv where u is the corresponding column in U for a user and v is the column in V for an item.

2.4.5 Expressiveness of the Minimax Entropy Model – Word2Vec Skip-Gram Model as A Special Case

The word2vec model and applications by Mikolov *et al.* [38] have attracted a great amount of attention in recent years. It is interesting to note that the Skip-Gram model is essentially a minimax entropy model. The context vectors can be viewed as "weights" and the word vectors can be viewed as "latent features". There are no explicit features. The parametric form for each latent feature is the identity tranform of a single latent parameter. The learning task jointly learns the optimal feature representation together with the model weights, as illustrated in Figure 2.1.



Joint learning of weights and features

- Skip-Gram Word2Vec model
  - Word vectors $\{\mathbf{\Theta_1}, \mathbf{\Theta_2}, \dots\}$
  - Context vectors $\{\mathbf{w_1}, \mathbf{w_2}, \dots\}$
  - Vocabulary size: V
- Corresponding minimax entropy model:
  - Feature vector for (word x, context y) pair:
    - $\mathbf{f}(x,y,\mathbf{\Theta}) = [\overbrace{\mathbf{0}, \dots, \mathbf{0}}^{y-1}, \mathbf{\Theta}_x, \overbrace{\mathbf{0}, \dots, \mathbf{0}}^{V-y}]$
  - Global weight vector: $\mathbf{w} = [\mathbf{w_1}, \mathbf{w_2}, \dots]$

Figure 2.1: Model Expressiveness

2.4.6 Connection to Deep Learning

Deep learning algorithms [4, 14] attempt to model high-level abstractions in data by using multiple processing layers, with complex structures or otherwise, composed of multiple non-linear transformations. Research in this area attempts to make better representations and create models to learn these representations from large-scale unlabeled data. Our model also aims to learn an optimized feature representation, but focuses on the scenario where domain knowledge/expertise are available. Instead of learning the feature representations completely from scratch, these prior knowledge are exploited to specify latent features with certain parametric forms. This approach is particularly valuable when a relatively restricted amount of data is available.

2.4.7 Joint Learning vs. Combining Fine-Tuned Ad-Hoc Models

The parametric form of a latent feature can be as simple as the identity transform. It also can be defined by a complex function involving a couple of parameters. The minimax entropy model takes a joint learning approach where all the parameters are jointly learned with feature weights. In contrast to approaches with solely explicit features (where latent parameters are either specified by human, or learned and fixed before learning feature weights), our strategy searches in a larger space (the full space spanned by feature weights and latent parameters). Although theoretically, a larger search space always produces better global optimum, in practice, the optimization procedure is usually not guaranteed to reach the global optimum due to the difficulty in exact optimization for non-convex objectives. The enlarged search space may result in a harder optimization problem, thus it is possible that the final solution turns out to be worse due to insufficient optimization. In view of such practical concern, we carefully designed the learning algorithm. The algorithm is initialized by the optimized parameters that we can obtain for individual features (See Algorithm 1[12]). The blocked coordinate descent routine is then activated. Since any traditional approaches can be utilized for the initialization, and the coordinate descent routine is guaranteed to yield a better solution, the final solution will not be worse than an ad-hoc model with fixed parameters.

[12] K-means clustering is employed for the initialization in this case. We can use more sophisticated models as well.

2.4.8 Cold Start

As in most recommendation problems, cold start is an important issue in preference learning. If we have little historical data for a user, predicting her preference typically falls back to an appropriate way of utilizing ”independent” features that do not reply on histories, such as gender, age, hometown, etc. Our model can elegantly handle such cases by just taking care of these information as additional features. They can be both explicit or latent. In this study, we do not have those demographic information available thus we do not define them in the profiles. However, these features can be utilized exactly the same way as the defined ones. We don’t even need to worry about how to distinguish cold start users from the heavy users since the automatically learned weights help us to do the trade-off. In the extreme case where a user has no history at all, the prediction will fall back to a regression on those ”independent” features. This treatment of cold start scenarios is of the similar style as in [1], with better expressiveness, reduced model complexity and simpler optimization procedure.

2.5 Experiments

We introduce our datasets and report our experimental results in this section. We evaluate our proposed method on the location prediction/recommendation task.

First we evaluate the effectiveness of our method by accuracy of prediction under various settings. Then we zoom in to see the benefits from optimizing the latent parameters. At the end we conduct an efficiency study and analyze the scalability of our method.

2.5.1 Data

We conduct our experiments on two public real world datasets [17] obtained from Foursquare[13]. The first dataset (**CA**) contains $483,813$ check-in records of $4,163$ users in California USA ranging from December 2009 to June 2013. The second dataset (**World**) contains $2,290,996$

[13] https://foursquare.com

check-in records of $11,326$ users around the world ranging from January 2011 to December 2011. We preprocess the datasets by removing the users and POIs with check-ins fewer than 20. Each check-in record consists of a user ID, a POI ID, a check-in timestamp, and the latitude and longitude of the POI. The first dataset has the category information of each POI while the second one does not.

We sort each user's check-ins chronologically and assign the first 80% of the check-ins to the training set and the remaining 20% to the test set.

2.5.2 Implementation

Smoothing the Category and Temporal Preference

Smoothing is a common practice to avoid overfitting and mitigate the effect from noise when estimating categorical distributions. It assigns a tiny probability to the categories that are not seen in the data. In our model, we do a simple add-one smoothing to the category preference and temporal preference of users.

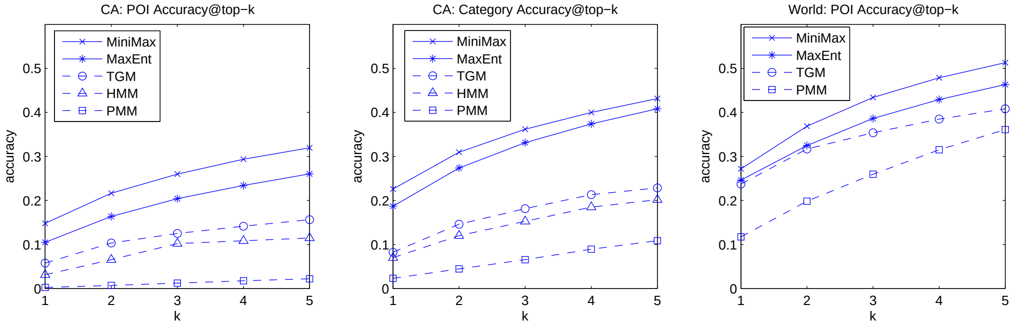


Figure 2.2: Prediction Accuracy @ top-k

Parameter Regularization

We incorporate a standard L_2 regularization on $\mathbf{w}$ in the MaxEnt step to avoid overfitting and numerical problems. The objective function becomes $-LL + \frac{1}{2}\beta||\mathbf{w}||_2^2$ where β is the regularization parameter. From our experiments, we found that our model is insensitive to parameter regularization. We set β to 0.2.

Number of Geographic Clusters

The number of clusters affect the granularity of the geographic regions. It can be empirically set by cross validation or specified by human knowledge of how fine grained regions we want to achieve. In this study, we set the cluster number to 30 for the CA dataset and 200 for the World dataset.

Number of Iterations

We set the global iteration number $Maxiter$ to 20 and run 10 iterations within each L-BFGS step based on the empirical study of the convergence rate.

2.5.3 Effectiveness Study

Methods for Comparison

Existing models on location prediction/recommendation are usually specifically designed emphasizing a particular set of factors. Unlike our model, most of them cannot be generalized to take arbitrary features. In this study, we consider category, time, popularity and geographic coordinates. Thus we compare our method (the basic **MaxEnt** model with K-means initialization, and the full **Minimax** model) with the following three state-of-art models which can accept the same set of features.

- **PMM**. A spatial-temporal location *prediction* model proposed in [11], which studies the spatial-temporal regularity of user mobilities and builds a generative model for check-ins.

- **HMM**. A mixed hidden Markov location *prediction* model proposed in [10], which first predicts the category of user activity at the next step and then predict the most likely location given the estimated category distribution. This model is compared to only for the CA dataset because the category information is not available for the World dataset.

- **TGM**. A time-aware location *recommendation* model proposed in [64], which employs a user-based collaborative filtering framework with geographic influence incorporated by a linear combination. For the CA dataset, we enhance this model by a further linear combination with the category distribution at the prediction time for fair comparison.

Evaluation on Accuracy

Evaluation Metrics. We compute the accuracy of both location and category prediction on the test set for the CA dataset and the accuracy of location prediction for the World dataset. For each $\langle utl \rangle$ in the test data, we return the top-k locations predicted by each model for (u, t). As long as the true location l lies in the top-k set, we consider it as a correct prediction. For categories, we obtain the category list associated with the top-k predictions and evaluate the accuracy in the same way.

Performance. As shown in Figure 2.2, our method significantly outperforms the three baselines w.r.t both POI and category prediction at all position k's. TGM is not working well because 1) it takes a binary user-location matrix as the input for collaborative filtering which completely ignores the preference over different visited POIs; 2) it involves geographic and temporal influences in an ad-hoc manner which is difficult to coordinate in the optimal way; 3) the way it encodes the geographic knowledge is to do a power-law fitting of consecutive check-in distances, which is sensitive to outliers and cannot capture the clustering effect of check-ins. HMM relies a lot on the Markovian assumption of user activity. If a user's check-ins are not so dense (which is usually the case since people do not check-in at every POI they visit), the dependency between consecutive check-ins are weakened. Once the Markovian assumption does not hold, good performance would not be guaranteed. PMM gives the worst performance. The generative assumption that movements are governed by Gaussian spatial-temporal clusters is too strict and limits the model's expressiveness and generalizability. Another interesting phenomenon we can observe is that despite the lack of category information for the World dataset, the location prediction accuracy is higher than the CA dataset for all the models. In fact, the World dataset has comparative number of POIs with the CA dataset but has substantially large number of checkin records. This makes the learning task easier for all the models. The performance difference is more significant on the CA dataset, which concludes that when we have limited number of observations for training, our MiniMax model generalizes better than the baseline models.

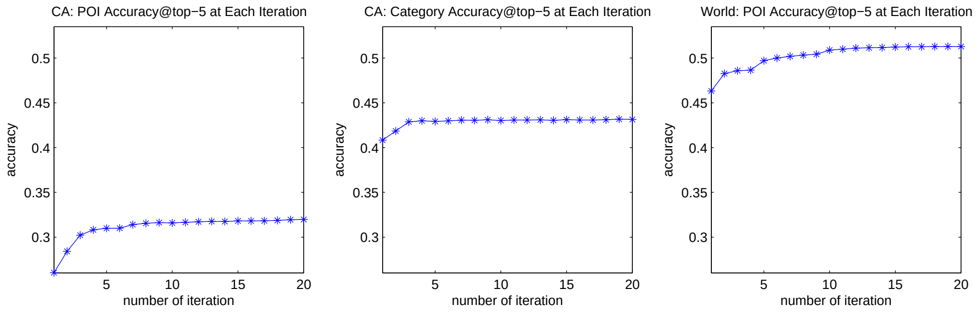
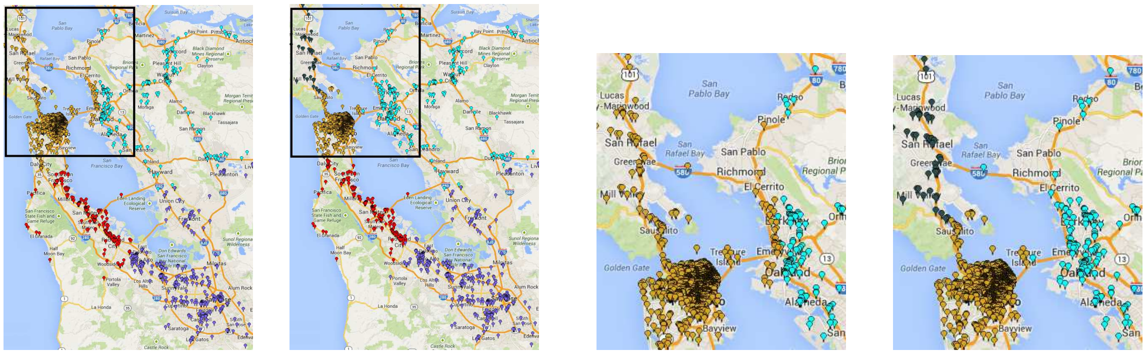


Figure 2.3: Prediction Accuracy @ top-5 as we optimize the latent parameters. The prediction becomes more accurate and the convergence is very fast.



(a) SF k-means (b) SF (c) SF k-means (d) SF

Figure 2.4: Geographic Clusters

The Influence of Latent Features

Performance. We plot the accuracy@top-5 of our model as the latent parameters are optimized with 20 iterations in Figure 2.3. The accuracy continues to improve as the latent parameters are optimized. It is also worth noting that the convergence is very fast.

A Visualization of the Geographic Clusters. To illustrate the intuition behind optimizing the latent parameters, we show a snapshot of the San Francisco Bay Area geographic clusters obtained from our algorithm for the CA dataset in Figure 2.4(b). We assign each POI l to a cluster by selecting the largest weight of $\mathbf{g}^l$. Figure 2.4(a) shows the initial k-means clustering results.

The optimal clustering structure is refined from the K-means clustering via the interaction with the check-in preferences modeling. We can observe interesting refinements. As shown in Figure 2.4(d) and Figure 2.4(c), we zoom in to San Francisco (SF) city. As K-means clustering blindly clusters the POIs by geographic latitudes and longitudes, the cluster centered at SF (yellow) stretched to San Rafael, Oakland and Berkeley; while in the refined clusters, SF corresponds to a concentrated cluster. The SF cluster extends north right to the vicinity of the Golden Gate Bridge as tourists to SF would always like to explore the Golden Gate Bridge.

2.5.4 Efficiency Study

In this section, we first analyze the complexity of our algorithm and then present experimental results on the execution time.

Complexity Analysis

The coordinate descent algorithm contains a MaxEnt step and a MinEnt step. Within each step, the space and time consuming part lies in the evaluation of the function value and the gradient (see Appendix B), which determines the complexity of our algorithm. We show that both space and time complexity are linear w.r.t the number of users, time indices and POIs.

Space. At each iteration of both steps, we need to store π_{utl} for all (u,t) pairs that appear in the training set and any $l \in L$, which requires at most $O(|U||T||L|)$ space. Computation of the feature values are done at the beginning of each step and requires at most two components of (u,t,l), therefore does not affect the order of space complexity. The space required to store the current estimate of the solution in the MaxEnt step is the dimension of the features $\mathbf{f}$. In the MinEnt step it is 3 times the number of geographic clusters, which is also not contributing to the order of complexity. Thus the overall space complexity is $O(|U||T||L|)$.

Time. At each iteration, to evaluate the function and gradient values, we need to compute π_{utl} for all (u,t) pairs that appear in the training set and any $l \in L$. Let the total iteration number be M and let the maximum function evaluation number be M_1 at one MaxEnt step and M_2 at one MinEnt step. The overall time complexity is $O(M(M_1 + M_2)|U||T||L|)$.

Execution Time Evaluation

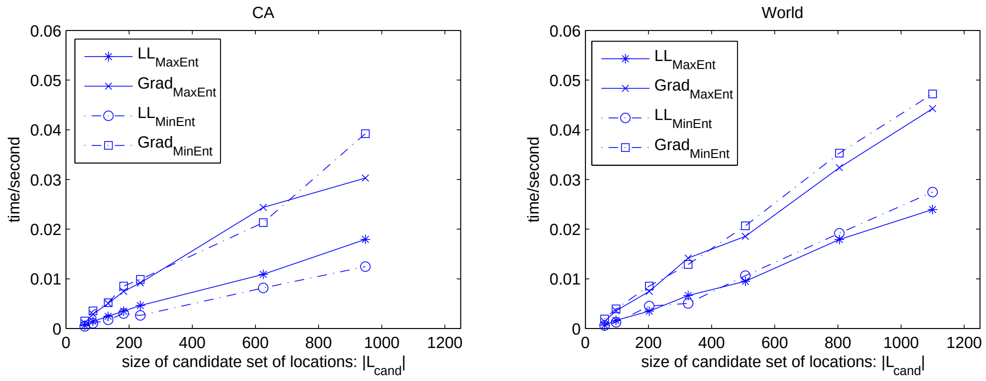


Figure 2.5: Average Execution Time of A Function/Gradient Evaluation

To examine the efficiency of our algorithm, we illustrate the execution time of a function/gradient evaluation for both the MaxEnt step and the MinEnt step. The time consuming computation of π_{utl} can be computed in parallel since $\{\pi_{utl}|l \in L\}$ can be computed for each (u, t) pair simultaneously. Therefore we examine the average execution time of a function/gradient evaluation over all (u, t) pairs. We vary the pruning threshold δ_α and obtain the *time - candidate set size* curves shown in Figure 2.5. They all exhibits a linear trend in $|L|$ while the gradient evaluation is more expensive than the function value evaluation. $|L_{cand}|$ is the average size of the candidate set over all (u, t) pairs. In the ideal case, the overall time complexity can be reduced to $O(M(M_1 + M_2)|L_{cand}|)$.

2.6 Summary

In this chapter, we develop a novel minimax approach for modeling time-aware check-in preferences. Specifically, our approach has the advantage of investigating the multidimensional knowledge of entities (users, locations) as well as jointly learning the latent geographic clustering. The proposed discriminative model can strike a good balance between explaining seen data and generalizing to unseen data by requiring the model to satisfy meaningful relaxed constraints. Going beyond check-in preference modeling, the proposed minimax entropy model also provides a general guidance to model ambiguous features with arbitrary parametric forms, which significantly boosts the flexibility and expressiveness of the standard discriminative learning models.

Chapter 3

Mining Multi-Aspect Reflection of News Events in Twitter

3.1 Overview

Social media data are unstructured, fragmented and noisy. It makes social media data mining even more challenging that a lot of real applications come with no available annotation in an *unsupervised setting*. In this chapter, we study the practical problem of event detection and summarization in social media, taking a novel angle where external sources such as news media and knowledge bases are exploited to provide supervision.

A major event usually has repercussions on both news media and social media sites such as Twitter. Unlike the "free-style" social media posts, news articles are written in formal languages, concentrated on important facts, and have a broad coverage of major events. These properties make news an ideal source for guiding knowledge discovery in social media.

Once an influential event takes place, mainstream news media immediately react to it. News reports deliver real-time status of the event, covering every aspect with fairly standard languages. Informed by these reports, people post their opinions/comments and raise discussions on the event via microblogging sites such as Twitter. The different natures of these two sources provide a complementary view of an event: A reasonably objective and comprehensive presentation of an event, and a view full of opinions and sentiments from the public. Linking them together to provide a complete picture of an event can be of great interest to both policy makers and ordinary people seeking information.

Preliminary research towards this direction include [24], which finds the most relevant news articles to enrich a given tweet; and [51], which retrieves related social media utterances to a given news article. However, either a single tweet or a single news article has limited expressing power, even if the original piece of information is enriched by the retrieved counterpart.

In this chapter, we take a global perspective and offer event level summaries of both sources simultaneously. Consider a newly inaugurated mayor who would like to know what the public opinions are about major events in the past two weeks. The following capabilities are desirable: 1) What are the major events; 2) who are the key players in each event; 3) how people talk about each event; and 4) when is the event and how long does the event last?

In addition, we notice that a major event can have multiple aspects. For example, the `Sony Pictures Entertainment Hack`[1] event around December 2014 a) raises doubts on if North Korea is responsible for the hack; b) unexpectedly promotes the film "the Interview" and leads to a big success for its online release; and c) attracts attention from the White House. Each aspect has different focuses both in the sense of key players involved and the diverse public opinions. Therefore, the mining process should be able to distinguish different aspects for each event to present a holistic view.

To this end, we propose a unified framework for mining multi-aspect reflections of news events in Twitter. We aim to detect major events as well as the multiple aspects of each event. An aspect of an event is characterized by both a set of news articles which emphasize objective facts and a set of relevant tweets which contain rich opinions and sentiments. Using the previous example, aspect (b) of the `Sony Hack` event can be described by news articles with headlines like

"Sony plans a limited release for film on Christmas Day"

and tweets like

[1] http://en.wikipedia.org/wiki/Sony_Pictures_Entertainment_hack
We use `Sony Hack` in the rest of this chapter for brevity.

"I told y'all Sony would release The Interview. This has been the most shameless promotion for a film I've ever seen."

Challenges. We aim to address two major challenges while designing our model. First, we need to discover the "anchors" to link news and tweets. With a collection of news articles and huge numbers of random or informative tweets, the challenge is how to discover and formally define the interesting events and aspects, based on which to link the two sources. Second, the language models of news and tweets are drastically different. The linking process should be able to bridge the vocabularies between news and tweets as well as to accommodate different modeling of long text (news) and short text (tweets). While news-related tweets do share a critical proportion of keywords with the news articles, there exists non-negligible vocabulary gap between them [25].

Anchor Discovery. In our proposal, anchor discovery is achieved by a comprehensive study of news solely instead of mixing these two sources at the early stage, in light of the high quality, less noise and broad coverage of news articles. To learn the optimal representation of the news events and their multiple aspects, we propose a novel and efficient *generative model* with an elegant recursive *decomposition strategy* for dynamic hierarchical entity-aware event/aspect discovery. The hierarchical structure is illustrated in Figure 3.1. The root node denotes the entire news collection, from which events are learned. Each event has a number of child nodes which denote aspects of this event[2]. *The event/aspect discovery is essentially a top-down hierarchical clustering procedure which recursively applies the generative model.* Our proposed decomposition strategy (Section 3.4.3) complies with the fact that aspect nodes originate from the same theme of their parent event node, while each aspect has its distinct emphasis. The generative model integrates the most critical dimensions for clustering, including text, entities (person/location/organization) and time in a unified manner. These dimensions mutually reinforce each other to boost coherence. A node is characterized by a word distribution, a set of entity distributions (with respect

[2] Our algorithm allows each aspect to have sub-aspects as well.

to person, location and organization), and a time distribution. These distributions form an accurate multidimensional *descriptor* for an event/aspect comprehensively.

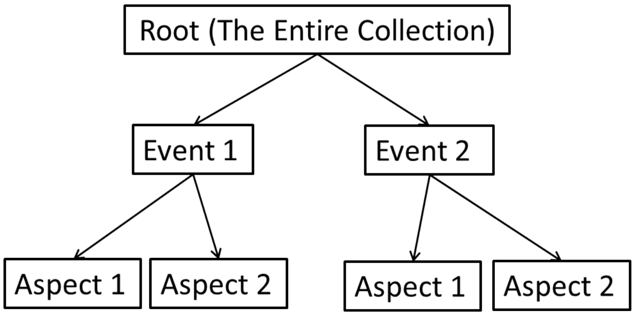


Figure 3.1: Event-Aspect Hierarchy.

Linking. The event/aspect descriptors are then utilized to guide the reflection mining. The goal is to investigate how various aspects of an event are discussed in Twitter. This is formulated as a bootstrapped dataless[3] multi-class classification problem [47]. Specifically, for each event, we first form a pool of candidate tweets out of the high-volume tweet stream by information retrieval with the multidimensional event descriptor. A retrieval model is proposed to retrieve tweets which achieve simultaneously textual, entity and temporal relevance to the event. Within the candidate pool, we use the aspect descriptors to select their corresponding initial confident sets of tweets (seeds). Then by bootstrapping we select and classify the candidates into different aspects until the number of tweets for each aspect meets a threshold. We can see that the entire process is unsupervised and no labeled data is required. Furthermore, the classifier is able to accommodate various local or global features. More significantly, the bootstrapping scheme not only benefits the classification accuracy itself, but also naturally handles the vocabulary gap between news and tweets.

Presentation. Aside from discovery and linking, how to present the well-sorted information to the end-users is non-trivial. For each aspect of an event, our framework naturally supports

[3] The name *labelless* classification may be more accurate and intuitive but we follow the terminology *dataless* due to historical reasons.

a user friendly presentation with an entity graph, a time span, a news summary and a tweet highlight for user digestion.

The last contribution of this work is the capability to create an aspect-specific and time-aware event dataset for an arbitrary time period, which prepares fine input for various applications such as opinion mining/comparison, multi-corpus text summarization and information diffusion.

The rest of this chapter is organized as follows. We state the problem in Section 3.2, followed by our proposed solution in Section 3.3. We present the key components of our solution, event/aspect discovery and tweets linking in Sections 3.4 and 3.5. We evaluate the proposed solution in Section 3.6, review related work in Section 3.7, and summarize this study in Section 3.8.

3.2 Problem Formulation

We formulate our problem in this section. The notations used in this chapter are summarized in Table 5.1.

Table 3.1: Summary of Notations

Symbol	**Description**
$\mathbf{X}^{\mathtt{w}}$	word matrix
$\mathbf{X}^{\mathtt{e}}$	entity matrix, $\mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o}$
$\mathbf{t}$	time vector
$\mathcal{I}$	input data associated with a node in the hierarchy. $\mathcal{I}_0$ denotes the root node, $\mathcal{I}_z$ denotes an event/aspect node
$\phi^{\mathtt{w}}$	word distribution
$\phi^{\mathtt{e}}$	entity distribution, $\mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o}$
μ, σ	parameters of time distribution
z	event/aspect ID;
	event/aspect descriptor: $z = \{\phi^{\mathtt{w}}, \{\phi^{\mathtt{e}}\}, \mu, \sigma\}$
θ	per-document topic distribution. The topic can be event or aspect

DEFINITION 5 (News Article)**.** *A news article is defined by a bag-of-words/entities model with a timestamp. The entities can be persons, locations or organizations*[4]*.*

A collection of news articles are thus compactly represented by 1) a $N^{\mathtt{w}} \times D$ word matrix $\mathbf{X}^{\mathtt{w}}$ where an entry $x^{\mathtt{w}}_{wd}$ denotes how many times the w-th word appears in the d-th news article; 2) three $N^{\mathtt{e}} \times D$ entity matrices $\{\mathbf{X}^{\mathtt{e}}\}$, where $\mathtt{e}$ can be the type person, location, or organization, *i.e.*, $\mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o}$. An entry $x^{\mathtt{e}}_{ed}$ denotes how many times the e-th entity appears in the d-th news article; 3) a time vector $\mathbf{t}$ where t_d denotes the timestamp of the d-th news article.

DEFINITION 6 (Tweet)**.** *A tweet is also defined by a bag-of-words/entities model with a timestamp.*

DEFINITION 7 (Event/Aspect)**.** *Events and aspects are nodes in a topically coherent hierarchy. Both an event node and an aspect node is defined by textual, entity and temporal dimensions. Formally, it is defined by 1) a multinomial word distribution $\phi^{\mathtt{w}}$; 2) a set of entity distributions $\{\phi^{\mathtt{e}}\}$, $\mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o}$, where $\phi^{\mathtt{p}}, \phi^{\mathtt{l}}, \phi^{\mathtt{o}}$ are all multinomial distributions; and 3) a Gaussian time distribution $\mathcal{N}(\mu, \sigma)$.*

DEFINITION 8 (Event/Aspect Descriptor)**.** *We denote an event/aspect descriptor by $z = \{\phi^{\mathtt{w}}, \{\phi^{\mathtt{e}}\}, \mu, \sigma\}$.*

DEFINITION 9 (Reflection)**.** *The reflection of an aspect of a news event is the set of relevant tweets to the aspect, which will be identified by the event and aspect descriptors.*

With the definitions above, we are now able to formulate our problem as follows.

PROBLEM 2. ***Event-Based Multi-Aspect Reflection Mining***

Given a collection of news articles and a collection of tweets within a query time period, learn the events during the period and the multiple aspects of each event; find the reflections in twitter; and present the multi-aspect events and their reflections to end users.

[4] Entities are extracted by NLP tools from the news content in a preprocessing step. See details in the Experiment section.

3.3 Overview of the Event-Based Multi-Aspect Reflection Mining Framework

To mine the reflections of multiple aspects of a news event, we propose a framework that can be divided into two main parts: event and aspect discovery in news, and linking with relevant tweets. Our process for event and aspect discovery in news involves a dynamic hierarchical entity-aware generative model with an elegant recursive decomposition strategy. After learning the accurate event and aspect descriptors via the generative model, we perform a bootstrapped dataless multi-class classification using the descriptors for identifying relevant tweets.

The goal of our generative model is to provide accurate descriptors for each event and aspect. The model leverages text, entities and time jointly in the generative process to enforce coherence through all these dimensions. The estimated distributions of words, entities and time form comprehensive event/aspect descriptors, which are the input for the following tweets linking part. For the construction of the event-aspect hierarchy, we propose a recursive decomposition strategy which naturally a) encodes the intuition that aspect nodes originate from the same theme of their parent event node, while each aspect has its distinct emphasis, b) supports a lazy learning protocol for efficient query processing: the aspects of an event are not learned until a user queries to expand the event.

Tweets are by nature noisy, informally written and filled up with all kinds of information. Identifying the relevant tweets discussing a particular aspect of an event is useful yet challenging. We address this by proposing a *retrieval + bootstapped dataless classification* procedure. For each event, with the event descriptor, we first devise a multidimensional retrieval model to retrieve an initial pool of tweets. Then with the aspect descriptors, we select informative tweets for each aspect iteratively by bootstrapping, which elegantly bridges the vocabulary gap between news and tweets. We expound upon our event/aspect discovery algorithm and tweets linking procedure in Section 3.4 and Section 3.5, respectively.

3.4 Event and Aspect Discovery in News

As discussed in Section 3.2, events and aspects are viewed as nodes in a topically coherent hierarchy. We propose a unified generative model for recursive construction of the hierarchy in a top-down manner. Essentially, it is a top-down hierarchical clustering process.

Step 1. Construct $\mathcal{I}_0 = \{\mathbf{X}^{\mathtt{w}}, \{\mathbf{X}^{\mathtt{e}}\}, \mathbf{t}\}$ using the entire collection of news. $\mathcal{I}_0$ is the input associated with the root node for inducing the event nodes.

Step 2. Induce the child nodes (events) of the root node taking $\mathcal{I}_0$ as input using the proposed generative model. The model estimates the descriptor $z = \{\phi^{\mathtt{w}}, \{\phi^{\mathtt{e}}\}, \mu, \sigma\}$ for each child node. We associate node z[5] with $\mathcal{I}_z$, which is generated by decomposing $\mathcal{I}_0$ to node z. Specifically, $\mathcal{I}_z = \{\mathbf{X}^{\mathtt{w}}_z, \{\mathbf{X}^{\mathtt{e}}_z\}, \mathbf{t}\}$, where $\sum_z \mathbf{X}^{\mathtt{w}}_z = \mathbf{X}^{\mathtt{w}}$, $\sum_z \mathbf{X}^{\mathtt{e}}_z = \mathbf{X}^{\mathtt{e}}$, $\mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o}$.

Step 3. Apply Step 2 to each event node z to induce the child nodes (aspects).

Recursively applying step 2 will further give sub-aspects, sub-sub-aspects and so on. Whether to split a node and how many child nodes to use depend on how "big" the current node is. We make this decision based on the logarithm of the amplitude of the matrices in $\mathcal{I}_z$. In this study, a two-level hierarchy is constructed, *i.e.*, the event level and the aspect level. However, our experiment system is implemented in a user-interactive fashion where users can decide the layout of the hierarchy with varying depth and branching numbers. We describe the key Step 2 (the generative model and the decomposition strategy for hierarchy construction) in the following sections.

3.4.1 The Generative Model

Our model assumes that each news article is generated by a mixture of topics (At the event level the topic denotes an event and at the aspect level it denotes an aspect.) governed by a multinomial topic distribution θ. The topic distribution is shared by each dimension, *i.e.*, text, entities and time. This is motivated by the intuition that all the above dimensions

[5] In this chapter, we slightly abuse the notation of z which is used both as a descriptor of a node and the ID of the node.

should be coherent for each topic: a news article is more likely to belong to a particular topic when its text, entities and timestamp all have high probabilities of belonging to the topic. For instance, a news article which contains words like *"film"*, *"release"*, entities like *"sony entertainment"*, *"Seth Rogen"*(the director of the film), and was published around *December 25, 2014*, would have high probability of belonging to the "film release" aspect of the Sony Hack event. Any single dimension is not sufficient for the conclusion.

Another important design of our model is to introduce a background topic B[6], which not only serves the traditional purpose of attracting the collection's aggregate characteristics for making other discovered topics more discriminative, but also enables an elegant decomposition strategy to construct the hierarchy. Under our decomposition strategy, we will see in what follows that the descriptor of the background topic for a set of nodes turns out to be exactly the descriptor of their parent node. In other words, the background topic of an aspect has the same representation with that of the corresponding parent event. This matches the intuition that a news article is a mixture of an event background topic and a set of aspect topics.

The plate notation for the generative model is shown in Figure 3.2. We observe words, entities and the timestamp for each news article and estimate the parameters

$$\mathbf{\Theta} = \{\{\theta\}, \{\phi^{\mathtt{w}}\}, \{\phi^{\mathtt{p}}\}, \{\phi^{\mathtt{l}}\}, \{\phi^{\mathtt{o}}\}, \{\mu\}, \{\sigma\}\}$$

The generative process is as follows:

To generate each word in news article d,

1. Draw a switch variable $s^{\mathtt{w}} \sim Bernoulli(\lambda_B)$. λ_B is the topic proportion of the background topic B.

[6] The background topic B is also defined by multiple dimensions with the collection's word distribution $\phi^{\mathtt{w}}_B$, the collection's entity distributions $\{\phi^{\mathtt{e}}_B\}$ and the collection's temporal distribution $\mathcal{N}(\mu_B, \sigma_B)$ where μ_B and σ_B are the mean and standard deviation of the collection's timestamps.

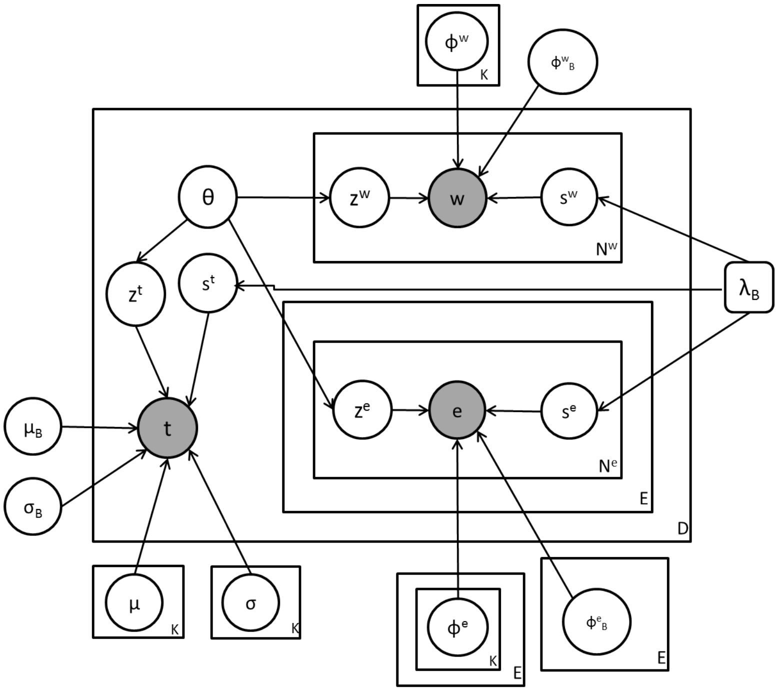


Figure 3.2: Plate Notation: News Learning Module

2. If $s^{\mathtt{w}} = 1$,

draw a word w from the background topic B: $w \sim \phi^{\mathtt{w}}_{B}$;

Else,

draw a topic $z^{\mathtt{w}}$ from the topic distribution θ_d,

draw a word w from the topic $z^{\mathtt{w}}$: $w \sim \phi^{\mathtt{w}}_{z^{\mathtt{w}}}$.

To generate a timestamp t_d for news article d,

1. Draw a switch variable $s^{\mathtt{t}} \sim Bernoulli(\lambda_B)$.

3. If $s^{\mathtt{t}} = 1$,

draw a timestamp t_d from the background time distribution B: $t_d \sim \mathcal{N}(\mu_B, \sigma_B)$;

Else,

draw a topic $z^{\mathtt{t}}$ from the topic distribution θ_d, draw a timestamp t_d from the topic $z^{\mathtt{t}}$: $t_d \sim \mathcal{N}(\mu_{z^{\mathtt{t}}}, \sigma_{z^{\mathtt{t}}})$.

For $\mathtt{e}$ in $\{\mathtt{p}, \mathtt{l}, \mathtt{o}\}$,

To generate each entity e in news article d,

1. Draw a switch variable $s^{\mathtt{e}} \sim Bernoulli(\lambda_B)$.

2. If $s^{\mathtt{e}} = 1$,

   draw an entity e from the background topic B: $e \sim \phi_B^{\mathtt{e}}$;

   Else,

   draw a topic $z^{\mathtt{e}}$ from the topic distribution θ_d,

   draw an entity e from the topic $z^{\mathtt{e}}$: $e \sim \phi_{z^{\mathtt{e}}}^{\mathtt{e}}$.

As shown in the above process, the posterior distribution of topics depends on the information from five dimensions – text, person, location, organization and time. Despite the fact that entities and time are by themselves interesting dimensions to describe each event/aspect, another important motivation to model them jointly with text is that they impose a regularization effect to the posterior distribution of topics which introduces mutual reinforcement among different dimensions.

3.4.2 Inference

We learn the parameters by Maximum Likelihood Estimation (MLE), searching the parameters that maximize the likelihood of the observations

$$\mathcal{L} = P(\mathbf{X}^{\mathtt{w}}, \{\mathbf{X}^{\mathtt{e}}\}, \mathbf{t} | \mathbf{\Theta}) \tag{3.1}$$

The objective function is thus

$$\begin{aligned}\mathbf{\Theta} =&\arg\max_{\mathbf{\Theta}}\mathcal{L}\\ =&\arg\max_{\mathbf{\Theta}}\alpha^{\mathtt{w}}\sum_{w,d}x^{\mathtt{w}}_{wd}\log\sum_{z}\phi^{\mathtt{w}}_{zw}\theta_{dz}+\\ &\sum_{\mathtt{e}}\alpha^{\mathtt{e}}\sum_{e,d}x^{\mathtt{e}}_{ed}\log\sum_{z}\phi^{\mathtt{e}}_{ze}\theta_{dz}+\\ &\sum_{d}\log\sum_{z}P(t_d|\mu_z,\sigma_z)\theta_{dz}\end{aligned} \tag{3.2}$$

To balance the influence from different dimensions, a tunable weight vector $[\alpha^{\mathtt{w}}, \alpha^{\mathtt{p}}, \alpha^{\mathtt{l}}, \alpha^{\mathtt{o}}, 1]$ is used to rescale the likelihoods [58], as is also common in speech recognition when the acoustic and language models are combined. The relative weight of text dimension to others determines the strength of the regularization effects. A natural setting is to allow α's to normalize the likelihoods from all the dimensions to the same scale.

We use the standard Expectation-Maximization (EM) algorithm that iteratively infers the model parameters $\mathbf{\Theta}$. The estimation of the topic distribution θ is given by

$$P(z|d) \propto \alpha^{\mathtt{w}}\sum_{w}x^{\mathtt{w}}_{wd}P(z|w,d) + \sum_{\mathtt{e}}\alpha^{\mathtt{e}}\sum_{e}x^{\mathtt{e}}_{ed}P(z|e,d) + P(z|t_d) \tag{3.3}$$

The first term resembles the estimation of the topic distribution in standard topic modeling, the second term integrates the entity dimensions, and the third term integrates the temporal dimension.

3.4.3 Hierarchy Construction

To construct the event-aspect hierarchy, we first apply our generative model to the entire collection $\mathcal{I}_0$ for event discovery. Then we decompose $\mathcal{I}_0$ based on the event descriptors to prepare input $\mathcal{I}_z$ for each event node z. The recursion begins at this point where we apply our generative model to each $\mathcal{I}_z$ for aspect discovery.

The key lies in an effective decomposition from $\mathcal{I}_0$ to $\mathcal{I}_z$. We outline the desired properties of the decomposition as follows.

- The word matrix and the entity matrices in $\mathcal{I}_z$ extract the portion of words/entities belonging to event z.
- The distributions in an event descriptor form the background topic descriptor of its child aspects.

The first property is intuitive. The second property is to ensure that aspects of an event originate from the same theme, while each aspect has its distinct emphasis.

We propose the following decomposition strategy based on the topic (event) membership of each word/entity in a document, which naturally embeds the above requirements.

$$\mathbf{X}^{\mathtt{w}}_z(w,d) = \mathbf{X}^{\mathtt{w}}(w,d) \times P(z|w,d) \tag{3.4}$$

$$\mathbf{X}^{\mathtt{e}}_z(e,d) = \mathbf{X}^{\mathtt{e}}(e,d) \times P(z|e,d), \mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o} \tag{3.5}$$

$P(z|w,d)$ denotes the posterior probability that the w-th word in the d-th document belongs to event z. Each entry $\mathbf{X}^{\mathtt{w}}(w,d)$ of the original word matrix $\mathbf{X}^{\mathtt{w}}$ is thus split to different events based on the posterior probability. The decomposition of entity matrices is done in the same way.

To see why the second property holds, let $(\phi^{\mathtt{w}}_B)_z, (\phi^{\mathtt{e}}_B)_z$ denote the background word and entity distributions computed with input $\mathcal{I}_z$, and let $\phi^{\mathtt{w}}_z, \phi^{\mathtt{e}}_z$ denote the word and entity distributions of event z estimated from the event discovery step. We have

$$(\phi^{\mathtt{w}}_B)_z(w) = \frac{\sum_d \mathbf{X}^{\mathtt{w}}_z}{\sum_{w,d} \mathbf{X}^{\mathtt{w}}_z} = \frac{\sum_d \mathbf{X}^{\mathtt{w}}(w,d)P(z|w,d)}{\sum_{w,d} \mathbf{X}^{\mathtt{w}}(w,d)P(z|w,d)} = \phi^{\mathtt{w}}_z(w) \tag{3.6}$$

$$(\phi^{\mathtt{e}}_B)_z(e) = \frac{\sum_d \mathbf{X}^{\mathtt{e}}_z}{\sum_{e,d} \mathbf{X}^{\mathtt{e}}_z} = \frac{\sum_d \mathbf{X}^{\mathtt{e}}(e,d)P(z|e,d)}{\sum_{w,d} \mathbf{X}^{\mathtt{e}}(e,d)P(z|e,d)} = \phi^{\mathtt{e}}_z(e) \tag{3.7}$$

The first equal sign in both equations follows by definition of the background topic. The second equal sign demonstrates our decomposition strategy. And the third equal sign follows from the updating formula in M-step of the inference procedure.

3.5 Tweets Linking

In the previous section, we have learned a word distribution, three entity distributions and a time distribution for each event and each aspect in the hierarchy. These distributions form comprehensive descriptors, which are used to find in Twitter the "reflection" of each aspect of a news event[7].

In this section, we first describe the candidate pool retrieval for each event with the event descriptor, and then elaborate the bootstrapping procedure which selects tweets for each aspect with the aspect descriptors.

3.5.1 Candidate Pool Retrieval with Event Descriptor

A candidate pool of tweets are retrieved for each event by information retrieval (IR). Specifically, we propose a language model which simultaneously investigate text, entities and time to determine the relevance of a tweet to an event.

The event descriptor is fed in as a query. Documents (tweets) are ranked by the probability of being generated by the query. This IR step is motivated by the fact that high volumes of tweets make it impossible to investigate every single tweet. The event descriptor provides a feasible way to retrieve a highly relevant candidate pool for identifying the reflections. The

[7] A substantial number of tweets contain a URL to a news article and the contents are just the news titles. Identifying these tweets are trivial in the linking task and do not add much value for users. In this study, we skip these cases and consider the tweets without URLs only.

score for ranking is derived as follows:

$$
\begin{aligned}
&\log P(d|z) \quad (d \text{ is a tweet and } z \text{ is an event descriptor}) \\
=&\alpha^{\mathtt{w}} \log P(d^{\mathtt{w}}|z) + \sum_{\mathtt{e}=\mathtt{p},\mathtt{l},\mathtt{o}} \alpha^{\mathtt{e}} \log P(d^{\mathtt{e}}|z) + \log P(d^{\mathtt{t}}|z)
\end{aligned} \tag{3.8}
$$

where $d^{\mathtt{w}}$ denotes all the words in d, $d^{\mathtt{e}}$ denotes all the type $\mathtt{e}$ entities in d, and $d^{\mathtt{t}}$ is the timestamp of d. The likelihoods from different dimensions are rescaled with α's by the same philosophy as in Section 3.4.2. Apply Bayes's rule to the first two terms as in standard query likelihood model, we obtain

$$
\begin{aligned}
&\log P(d|z) \\
\propto&\alpha^{\mathtt{w}} \log P(z|d^{\mathtt{w}}) + \sum_{\mathtt{e}=\mathtt{p},\mathtt{l},\mathtt{o}} \alpha^{\mathtt{e}} \log P(z|d^{\mathtt{e}}) + \log P(d^{\mathtt{t}}|z) \\
=&\alpha^{\mathtt{w}} \sum_{w} \phi^{\mathtt{w}}_{zw} \log P(w|d) + \sum_{\mathtt{e}=\mathtt{p},\mathtt{l},\mathtt{o}} \alpha^{\mathtt{e}} \sum_{e} \phi^{\mathtt{e}}_{ze} \log P(e|d) + \log P(d^{\mathtt{t}}|z)
\end{aligned}
$$

This is the final score used for ranking, where $P(d^{\mathtt{t}}|z) \sim t|N(\mu_z, \sigma_z)$, $P(w|d)$ and $P(e|d)$ are obtained by a Dirichlet smoothing to the language model of a tweet d:

$$
P(w|d) = \frac{\#(w,d) + \mu P(w)}{\#w + \mu} \tag{3.9}
$$

$$
P(e|d) = \frac{\#(e,d) + \mu P(e)}{\#e + \mu} \quad e = p, l, o \tag{3.10}
$$

3.5.2 Dataless Bootstrapping

We select and rank tweets for each aspect by a bootstrapped multi-class dataless classification scheme. We classify the tweets in the candidate pool into different aspects and select the top ones for each aspect. In addition to the multidimensional relevance requirement, this step is motivated by a) the existence of vocabulary gap between news and tweets; and b) the existence of noisy tweets which are irrelevant to any aspect.

Bootstrapping provides a way to weigh the semantic representation extracted from news that best fits the specific tweet collection. It starts with a confident seed set for each aspect obtained using the aspect descriptor. These are viewed as the first batch of labeled data. In each iteration, a multi-class classifier is trained using the current labeled data. And then the classifier labels more data by selecting the most confident tweets from the unlabeled ones. After each iteration, the accuracy of the classifier is improved and more labeled data are incorporated. The procedure is summarized as follows:

Step 1: Initialize M most confident seed tweets for each aspect using the aspect descriptors. The confidence is measured by the score from the language model as in Eq. (3.9).

Step 2: For each iteration, train a classifier based on the current set of labeled data and label N more tweets for each aspect.

Step 3: Repeat Step 2 until a desired number of tweets for each aspect are retrieved or the confidence score is lower than a threshold.

The classifier can be any multi-class classifier taking arbitrary features. In this study, we use logistic regression with L2 regularization. The features we use are listed as follows.

- Tf-idf word features. The values are scaled to range $[0, 1]$.

- Tf-idf entity features. The values are scaled to range $[0, 1]$.

- Time vector. For a tweet with a timestamp t, the i-th element in the time vector is the probability density at t computed using the time distribution of the i-th aspect. The vector is normalized to sum to 1.

3.6 Experiments

We perform empirical study to answer the following questions: 1) how effective is the event-aspect hierarchy learning? and 2) how well is the tweets linking quality? At the end, we demonstrate that our framework naturally supports a user friendly presentation with entity graphs, time spans, news summaries and tweet highlights.

3.6.1 Dataset Description

We consider two datasets in our experiments.

TopStory We crawled the top stories (full text) from Google News[8] every 30 minutes from Dec 20, 2014 to Jan 4, 2015. For each news, we query the Twitter Search API[9] with the extracted noun phrases from the title and snippet. Tweets containing at least one of the noun phrases are returned. We collected tweets that are posted within one day after the published time of the news. The dataset consists of $3,747$ news and $36,149,019$ tweets in total.

Link This dataset is provided by Guo *et al.* [24], which contains explicit URL links from each tweet to a related news article. They crawled $12,704$ CNN and NYTIMES news (title + snippets) from RSS feeds from Jan 11 to Jan 27, 2013. $34,888$ tweets that contain a single link to a CNN or NYTIMES news were collected during the same period. This dataset is a gold standard dataset to test the performance of the tweets linking module.

For both datasets, entities including persons, locations, and organizations are extracted using DBpedia Spotlight[10]. DBpedia is a project aiming to extract structured content from the information created as part of the Wikipedia project. This structured information is then made available on the World Wide Web. DBpedia allows users to semantically query relationships and properties associated with Wikipedia resources, including links to other

[8] https://news.google.com/

[9] https://dev.twitter.com/rest/public/search

[10] https://github.com/dbpedia-spotlight/dbpedia-spotlight

related datasets. [11] DBpedia Spotlight is a tool for annotating mentions of DBpedia concepts in plain text. It looks for 3.5M things of unknown or 320 known types in text and tries to link them to their global unique identifiers in DBpedia [12].

3.6.2 Implementation Details

For all the methods in our experiments, we set the number of iterations to be 20. The topic modeling parameters are initialized by the results from Kmeans clustering with 50 random initializations. Specifically, Kmeans is run on the tf-idf vectors of news articles. Topic distributions are initialized by the cluster assignments[12]. The word/entity/time distributions are initialized by the aggregate statistics of the documents in each cluster. The weights α's are tuned for each dataset on a develop set containing 1/10 of the dataset. Specifically, we first let α_0's to scale the likelihoods from different dimensions after the first iteration to the same value. Then we search in a grid centered at α_0's and select the configuration which leads to the highest pointwise mutual information (PMI) [40]. In the tweets linking procedure, we set $M = 50$ and $N = 10$.

3.6.3 Effectiveness and efficiency of the Event-Aspect Hierarchy Learning

We investigate the benefit from integrating multiple dimensions (entities and time) and compare with the state-of-art topical hierarchy construction method CATHYHIN [54]. Pointwise mutual information (PMI)[40] is used to measure the clustering quality, which is generally preferred over other quantitative metrics such as perplexity or the likelihood of held-out

[11] https://en.wikipedia.org/wiki/DBpedia

[12] For *e.g.*, if there are K clusters and document d is assigned to cluster 2, the topic distribution becomes $(s, 1-(K-1)s, ..., s)$. $s = 1/K^2$ is a smoothing parameter.

Table 3.2: Averaged PMI over Events using Top 20 Words from the Word Distributions

	TopStory	Link
30 events		
CATHYHIN	0.5239	0.2769
Text	0.702	0.2503
Text+Entities	0.7423	0.2803
Full	**0.773**	**0.2866**
150 events		
CATHYHIN	0.316	0.3123
Text	0.4065	0.2883
Text+Entities	0.4281	0.3151
Full	**0.4485**	**0.3222**

data[49]. We compare the average PMI over all events[13]. An efficiency analysis is presented at the end. Methods for comparison are summarized as follows.

- Our model with text dimension only;

- Our model with text + entity dimensions;

- Our full model with text + entity + time;

- CATHYHIN [54]. CATHYHIN takes a collection of documents and entities from a network perspective. They take the same input as our model and build the hierarchy recursively as well. But they work on networks formed by multiple combination of the matrix multiplications and conduct network clustering for topical hierarchy construction. For example, $\mathbf{X}^{\mathtt{W}} \times {\mathbf{X}^{\mathtt{P}}}^{T}$ forms a word-by-person network. CATHYHIN requires human to specify several types of networks and models the edge weight generation using a Poisson distribution. By default, all the entity type combinations are considered in the clustering process.

We list the results with two different number of events settings, *i.e.*, 30 events and 150 events. Similar results were observed for other numbers of events. As shown in Table 3.2, integrating

[13] The comparison is done for the event level because all the methods start with the same root node but the event clusters can be different which makes aspect level PMI incomparable.

entities and time increases the topical coherence. CATHYHIN has comparable performance with our Text+Entities model on the Link dataset but is significantly worse on the TopStory dataset. In fact, the Link dataset only contains titles and snippets which are of high quality. This makes the clustering task relatively easy. As CATHYHIN primarily relies on the co-occurrence matrices of all possible entity type combinations, it performs better on a smaller and cleaner dataset. Another significant observation is that our method is far more efficient than CATHYHIN since we work on the sparse document by words/entities matrices while CATHYHIN works on the co-occurrence matrices which is usually much denser especially for long text. Although we take the same amount of input knowledge, the running time of our method is in the order of several minutes but CATHYHIN takes several hours[14]. The running time of our method with varying event number is plotted in Figure 3.3. The results show that our model scales linearly with the event number. In fact, the complexity for each iteration of the inference process is dominated by the text dimension in the M-step, which is $O(K|\mathbf{X}^{\mathtt{w}}|)$, where K is the number of events and $|\mathbf{X}^{\mathtt{w}}|$ is the number of non-zero entries in the matrix. Thus our model scales linearly with the number of events and the size of the collection.

3.6.4 Tweets Linking

To quantitatively evaluate the linking procedure, we use the Link dataset which has explicit links between news and tweets. We compare with the WTMF-G method proposed in [24], which learns a latent representation (vector) for each news and tweet also considering multi-dimensional information such as text, entities and time. They use cosine similarity of the latent vectors to measure the relevance of a news and a tweet. The number of events is set to 150 because WTMF-G was reported to work best at this setting. We design the following experiment to study the precision and recall. Each news article d is assigned

[14] Both test are on a 16GB memory Matlab platform. For CATHYHIN, we used the implementation from the authors. CATHYHIN finishes in 3-4 hours for TopStory dataset and 10-20 minutes for Link dataset.

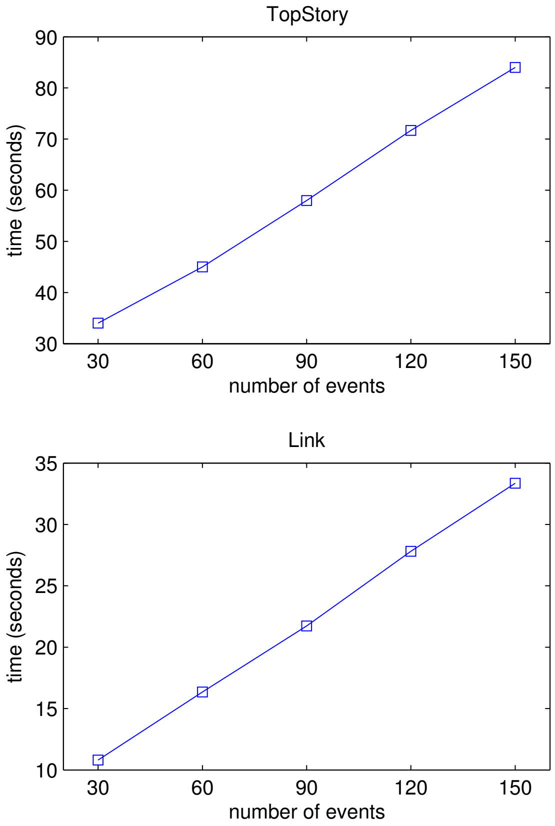


Figure 3.3: Running Time with Varying Event Number

to the event z^* by $z^* = \arg\max_z \theta_{dz}$. We take the top 20 events measured by the total number of news articles contained. For each of these events, our method select the top $k \times$ #(articles in the event) tweets. To compare with WTMF-G, we take the news assignments as given and consider two baselines derived from WTMF-G: 1) retrieve the top k tweets for each news article to form a same length of ranking list; 2) use the centroid of the latent vectors of the news in an event to retrieve $k \times$ #(articles in the event) tweets. We compute the average precision and recall for the top 20 events and randomly select one of them to evaluate the average precision and recall of its aspects.

The precisions/recalls are computed at the positions $1, 5, 10, 20$ and are plotted in Figure 3.4. Our method clearly outperforms both baselines. This demonstrates the effectiveness of our event/aspect descriptors and the bootstrapped dataless classification procedure.

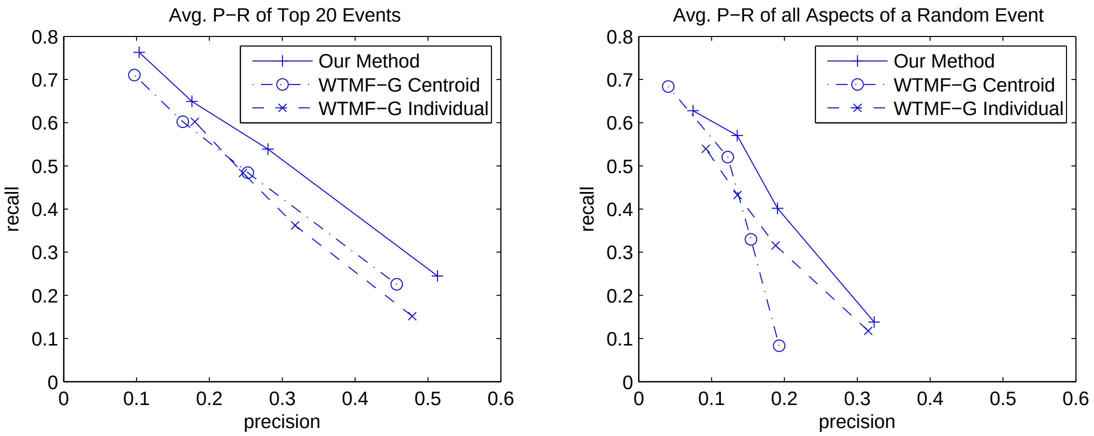


Figure 3.4: Precision-Recall Curves. The four points on each curve correspond to the precision/recall @ $1, 5, 10, 20$.

3.6.5 Presenting Results for User Digestion

It is always important yet challenging to present learned results to users in an informative way. Our framework naturally supports a user friendly presentation with entity graphs, time spans, news summaries and tweet highlights. We use the `Sony Hack` event in the Top-Story dataset to illustrate each component. The overall visualization of this event is given at Event_description_overall.html[15] where each aspect is given at Event_description_aspect1.html[16] (change 1 from 2 to 6 to see other aspects).

For each aspect of an event, we offer a view with an entity graph, a time span, a ranked list of news articles(the headlines are displayed) and a ranked list of tweets. We also offer an event view which integrates all the information of its aspects. In the following paragraphs, we explain how each component is generated. We use the `Sony Hack` event with a sample aspect about the "North Korea Internet Outage" as a running example.

An event $z = \{\phi_z^{\mathtt{w}}, \{\phi_z^{\mathtt{e}}\}, \mu_z, \sigma_z\}$ is associated with $\mathcal{I}_z$, which is used as input to discover aspects. Let $za = \{\phi_{za}^{\mathtt{w}}, \{\phi_{za}^{\mathtt{e}}\}, \mu_{za}, \sigma_{za}\}$ be the descriptor of the a-th aspect in event z, and let $\mathcal{I}_{za} = \{\mathbf{X}_{za}^{\mathtt{w}}, \{\mathbf{X}_{za}^{\mathtt{e}}\}, \mathbf{t}\}$ associate with node za.

[15] The text should be clickable. If not, go to `https://dl.dropboxusercontent.com/u/155956218/Event_description_overall.html`

[16] `https://dl.dropboxusercontent.com/u/155956218/Event_description_aspect1.html`

Entity Graphs

The recursive hierarchy construction leads to a natural visualization of the entity graph. For an aspect a in event z. The edge weight matrix $\mathbf{W}_{za}$ is given by

$$\mathbf{X}all_{za} = \text{vertical stack of}(\mathbf{X}^{\mathtt{p}}_{za}, \mathbf{X}^{\mathtt{l}}_{za}, \mathbf{X}^{\mathtt{o}}_{za}) \tag{3.11}$$

$$\mathbf{W}_{za} = \mathbf{X}all_{za}\mathbf{X}all^{T}_{za} \tag{3.12}$$

and the node weight is given by $\{\phi^{\mathtt{e}}_{za}\}$. For an event, an entity graph is constructed by combining all of its aspect entity graphs to form a multigraph, *i.e.*, two entities can be connected by multiple edges denoting their interaction in multiple aspects. The edge weights are the same as in individual aspect graphs while the node weights are given by $\{\phi^{\mathtt{e}}_{z}\}$. We give each aspect a unique color and let the node size (edge width) be proportional to the corresponding weight of a node (an edge).

The entity graph of the `Sony Hack` event is shown in Figure 3.5. Each node denotes an entity where the entities of the same type are in the same color. Each edge denotes the correlation between two entities where different colors represent the correlations in different aspects. We can see that *"Sony", "North Korea", "Kim Jong-un","Barack Obama", "Seth Rogen" and "James Franco"* are most influential in this event. If we zoom into the view of the red aspect, as shown in Figure 3.6, we can examine the entities in this particular aspect.

Time Spans

We use the Gaussian parameters μ_{za}, σ_{za} to generate the time distribution of each aspect. The time spans of different aspects in this event are shown in Figure 3.7, where the colors are consistent with the edges in the entity graph.

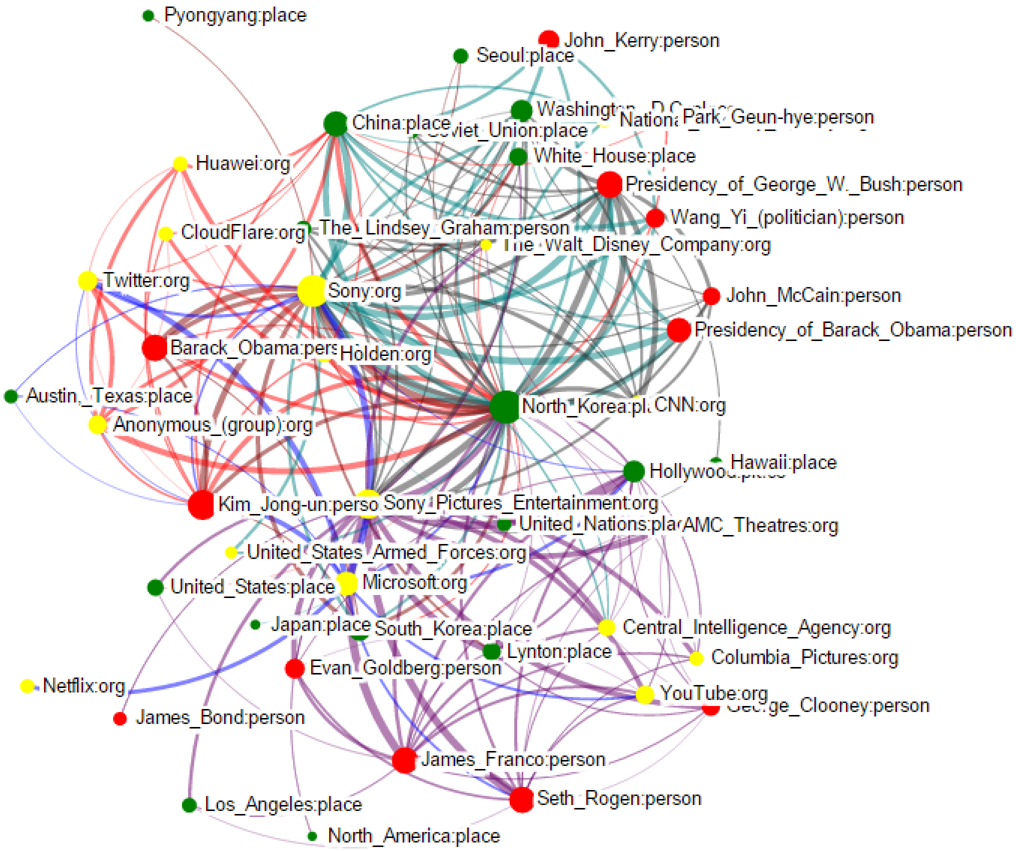


Figure 3.5: Entity Graph in the Event View. Red node: person; Green node: place; Yellow node: organization. The size of the node denotes the influence of the entity in this event. The width of the edge denotes the strength of the correlation between two entities. Different colors of edges represent the correlations in different aspects. We can see the influential entities in this event are: *"Sony", "North Korea", "Kim Jong-un","Barack Obama", "Seth Rogen (director and actor of the film)" and "James Franco (actor of the film)"* .

News Summaries and Tweet Highlights

While sophisticated news summarization can be performed to extract news summaries and tweet highlights, in this visualization we adopt a simple strategy. For the aspect a in z, we rank news articles by their posterior weight on a $P(a|d) = \theta_{da}$. We list the top five news articles in Table 3.3. Tweets are ranked by the output score of the classifier and we list the top five tweets together with the news summaries. The top five keywords from the word distribution are also listed. Obviously, the summaries, highlights together with the entity graph and the time span are of great help in understanding this aspect.

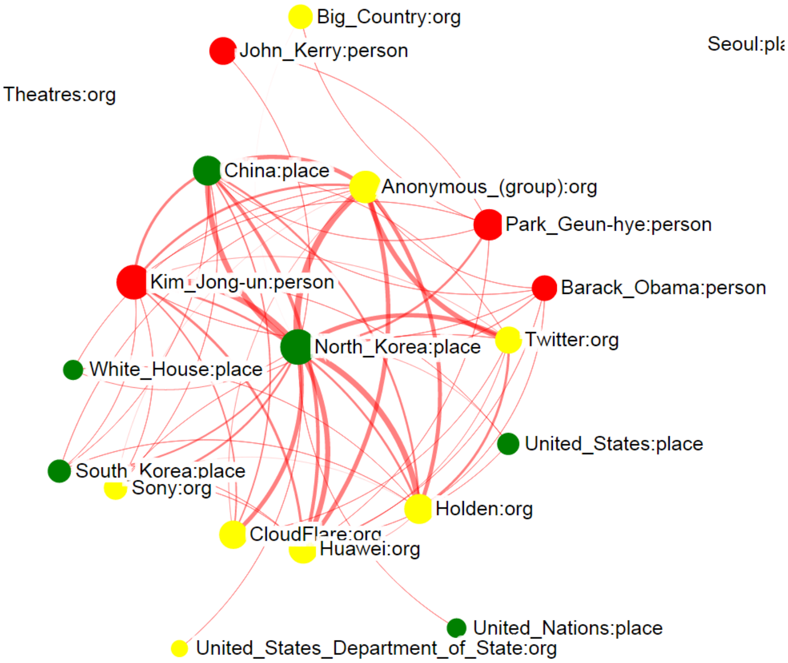


Figure 3.6: Zoom in to the Entity Graph of the Red Aspect about "North Korea Internet Outage". The size of a node denotes the influence of the entity in the aspect. The width of an edge denotes the strength of the correlation between two entities.

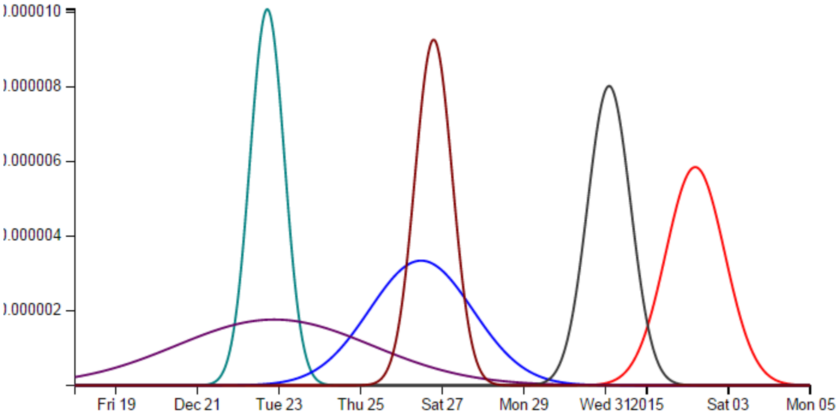
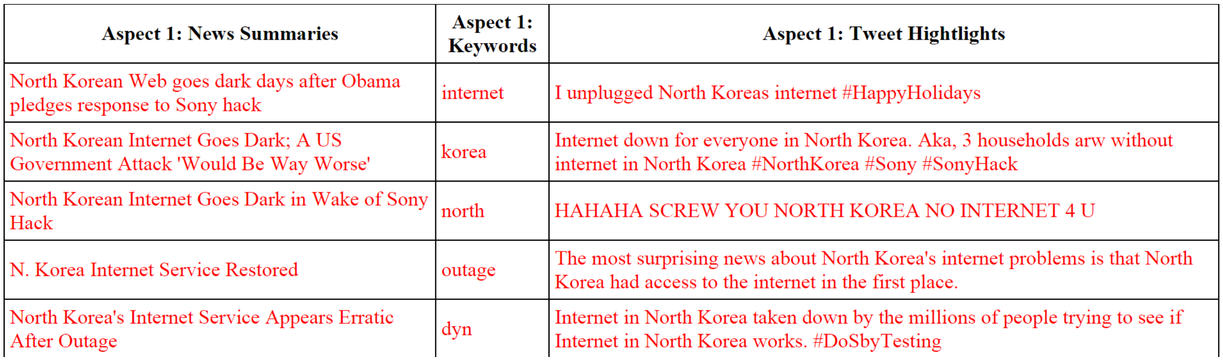


Figure 3.7: Time Spans in the Event View. The colors are consistent with the edges in the entity graph.

3.7 Related Work

To the best of our knowledge, this is the first work addressing the task of event based multi-aspect linking between news and tweets. Yet our work is related to topic modeling, event detection and several joint studies of news media and social media. In this section, aside

Table 3.3: The News Summary, Keywords and the Tweet Highlight of the aspect “North Korea Internet Outage” in the `Sony Hack` Event



Aspect 1: News Summaries	Aspect 1: Keywords	Aspect 1: Tweet Hightlights
North Korean Web goes dark days after Obama pledges response to Sony hack	internet	I unplugged North Koreas internet #HappyHolidays
North Korean Internet Goes Dark; A US Government Attack 'Would Be Way Worse'	korea	Internet down for everyone in North Korea. Aka, 3 households arw without internet in North Korea #NorthKorea #Sony #SonyHack
North Korean Internet Goes Dark in Wake of Sony Hack	north	HAHAHA SCREW YOU NORTH KOREA NO INTERNET 4 U
N. Korea Internet Service Restored	outage	The most surprising news about North Korea's internet problems is that North Korea had access to the internet in the first place.
North Korea's Internet Service Appears Erratic After Outage	dyn	Internet in North Korea taken down by the millions of people trying to see if Internet in North Korea works. #DoSbyTesting

from the related work we mentioned previously, we review the recent literature and make connections with them.

3.7.1 Topic Modeling

There has been a substantial amount of research on topic modeling. Inspired by entity topic models [27, 39], dynamic topic models [6, 58] and hierarchical topic models [7], we tailor our model to integrate multi-dimensional information for event/aspect learning. We follow the universal document-topic-word philosophy. In the mean time, we integrate entities and temporal information to jointly describe an event/aspect as well as to regularize the topic distributions. The proposed decomposition strategy provides a natural way for efficient hierarchy construction. Our model also provides an effective presentation for both user digestion and the tweets linking task afterwards. Provided the evidence by Masada *et al.* [37] that no meaningful difference between LDA and pLSI are observed for dimensionality reduction in document clustering, we intentionally leave out the prior for document-topic distributions as in LDA but take a pLSI style for an efficient EM optimization procedure, which is critical in hierarchical inference once the document collection becomes large. It is worth noting that our topic modeling algorithm scales linearly with the number of events and the length of the corpus.

3.7.2 Event Detection in Tweets

In the literature, there have been numerous research efforts aimed at event discovery in tweets [2, 44, 45, 46, 52], where various clustering methods taking well-calibrated features have been proposed. These studies focused on the single collection of tweets where huge number of random posts irrelevant to any news events interfere as noise. Our task distinguishes itself from this line of work by taking an opposite perspective. We discover events by investigating news articles, carefully learning different aspects and identifying their reflections in tweets, which is a more targeted and fine-grained task.

3.7.3 Joint Study of News Media and Microblogging Sites

Joint studies of news media and microblogging sites have attracted much attention recently due to a broad spectrum of potential applications. Zhao *et al.* [66] conducted a comparative study on the high level categories (politics, sports, etc.) and types (event-oriented, long-standing, etc.) of topics discovered from News and Tweets by running separate topic models in the two sources. Subavsic and Berendt [48] performed a case study to investigate text/headline/sentiment/entity divergence between news and tweets in an aggregate sense, concluding that a major role of Twitter authors consists of neither creating nor peddling, but extending them by commenting on news, which justifies the significance of our work. Gao *et al.* [22] studied the sentence level complementary news sentences and tweets and Wei and Gao[59] studied news highlights extraction utilizing tweets for a given event which can benefit from our event detection and representation. Within an event, Gao *et al.* [22] modeled dimensions such as location and time as latent aspects which were also characterized by word distributions, while they disregarded topical aspects. In our work we explicitly extract the entities from these dimensions, model them directly and go beyond events to find fine-grained topical aspects. Kothari *et al.* [28] and Masada *et al.* [53] utilized various features to classify tweets into comments or non-comments. These features can be well integrated to our classifier for tweets linking as well.

3.8 Summary

In this chapter, we proposed a unified framework to mine multi-aspect reflections of news events in Twitter. We proposed an effective time and entity-aware event/aspect discovery model to learn accurate descriptors of news events and their multiple aspects; the aspects of an event are linked to their reflections in Twitter by a bootstrapped dataless classification scheme, which elegantly handles the challenges of selecting informative tweets under overwhelming noise and bridging the vocabulary gap between news and tweets. Experimental results demonstrated that our framework can effectively retrieve the relevant tweets for fine-grained aspects of news events. While the scope of this chapter is to accurately identify the “reflections” of news events in twitter, discovering new aspects in Twitter which are not emphasized in news is an interesting future direction. We also demonstrated that our framework naturally generates an informative presentation of each event with entity graphs, time spans, news summaries and tweet highlights to facilitate user digestion. The capability of creating a high-quality aspect-specific and time-aware event dataset is of considerable practical benefits for various interesting applications such as comparative opinion mining and multi-corpus text summarization.

Chapter 4

Demo: EKNOT - Event Knowledge from News and Opinions in Twitter

4.1 Background

Massive information from news media and social media is more easily accessible than ever in this big data era. In this chapter, we develop a system named EKNOT which effectively discovers major events from news and connects each event to its discussion in Twitter. Essentially, EKNOT instantiates the technologies proposed in Chapter 3. It also provides guidelines on how to efficiently implement a joint event summarization module in practice. Given a time period, the system intends to answer the following questions: 1) What are the major events; 2) who are the key players in each event; 3) how do people talk about each event and what are their opinions; 4) when is the event and how long does the event last; 5) what are the multiple aspects (sub-events) if the event is rather big and influential? And what are the answers to the above questions for each aspect? EKNOT provides informative and comprehensive summaries for users to digest the huge amount of information effectively.

4.2 System Overview

Figure 4.1 illustrates the system architecture of EKNOT, which contains four major modules: data collection, event discovery, tweets linking and joint summarization. The input is a time period and the output is the summaries for the events and aspects.

EKNOT constantly crawls data from Google news. The key phrases extracted from each news title/snippet are used to query twitter API to obtain an initial pool of relevant tweets.

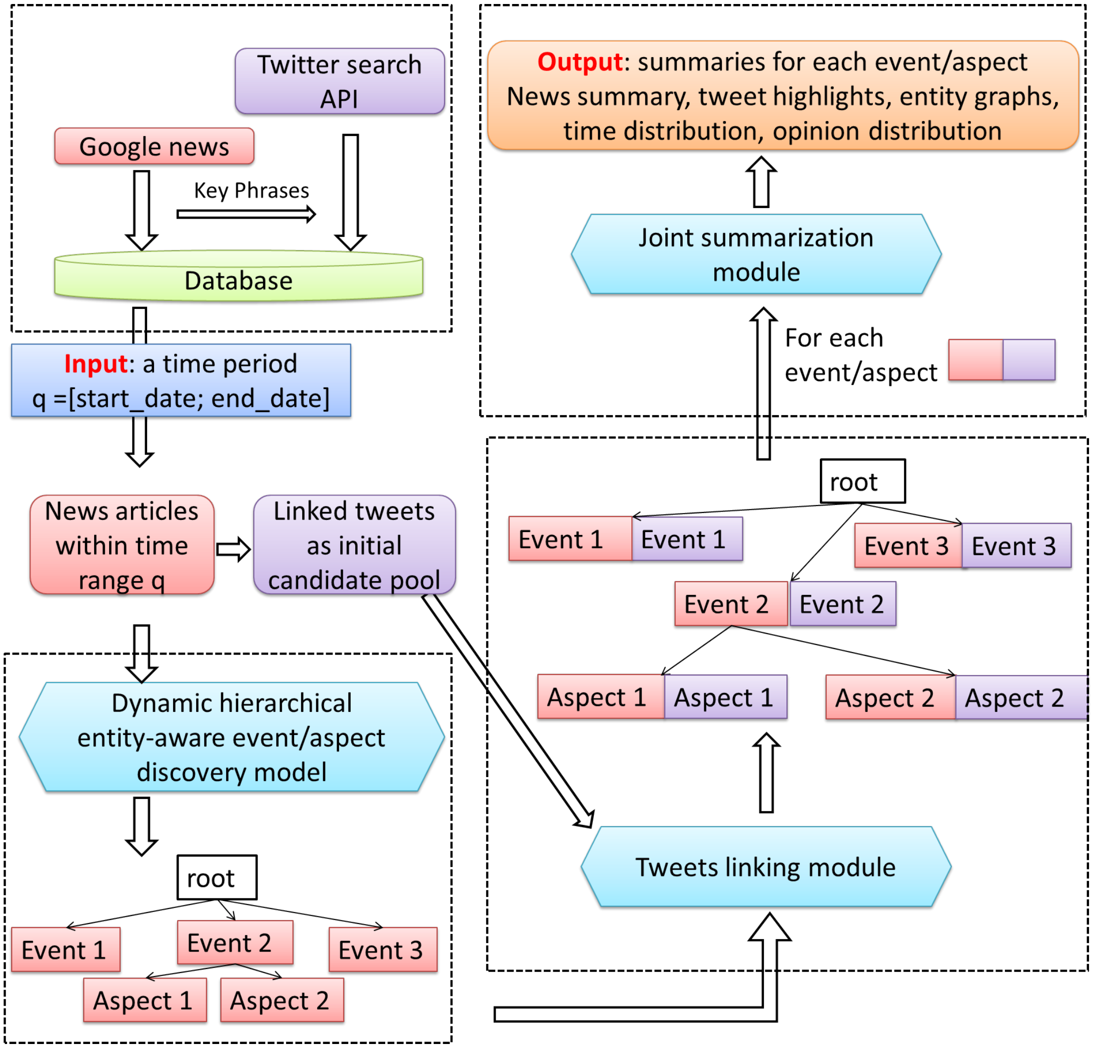

Twitter search API
Google news
Key Phrases
Database
Input: a time period
q =[start_date; end_date]
News articles within time range q
Linked tweets as initial candidate pool
Dynamic hierarchical entity-aware event/aspect discovery model
root
Event 1
Event 2
Event 3
Aspect 1
Aspect 2
Output: summaries for each event/aspect
News summary, tweet highlights, entity graphs,
time distribution, opinion distribution
Joint summarization module
For each event/aspect
root
Event 1
Event 1
Event 3
Event 3
Event 2
Event 2
Aspect 1
Aspect 1
Aspect 2
Aspect 2
Tweets linking module

Figure 4.1: System Architecture. Throughout this chapter, the red background indicates news side and the purple background indicates Twitter side.

Entities of the type person, organization or place are extracted from the news articles using NLP tools. Given a time period, EKNOT discovers major events from news based on the topic model proposed in the previous chapter, in light of the high quality and broad coverage of news articles. An event descriptor contains a word distribution, a time distribution and three entity distributions with respect to person, organization and place. The learned event descriptors are utilized to select the relevant tweets for each event and to analyze people's opinions from Twitter. At last, a joint summarization module leverages the descriptors, news articles and selected tweets to construct a news summary, tweet highlights, an entity graph, a time span and an opinion distribution for each event. Our event discovery module is instantiated by hierarchical topic modeling in a recursive manner, which allows users to zoom into a particular event interactively. Users can further investigate the event of interest and get the same style of summary for each aspect of the event.

4.3 Major Functional Modules

We describe the major functional modules of EKNOT in this section.

4.3.1 Data Collection

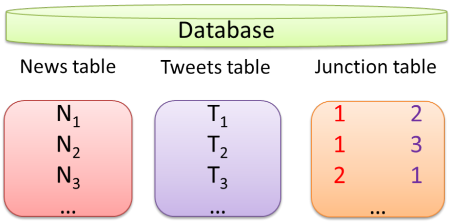


Figure 4.2: Database

EKNOT crawls the top stories from Google News[1] every 30 minutes. For each news article, it queries the Twitter Search API[2] with extracted noun phrases (by TextBlob[3]) and entities (by DBpedia Spotlight[4]) from the title and the snippet. Tweets posted within one day after the news and containing at least two of the noun phrases or entities are returned[5]. Our database consists of a news table, a tweets table and a junction table recording the many-to-many mapping between a news article and a tweet, as illustrated in Figure 4.2.

[1] https://news.google.com/

[2] https://dev.twitter.com/rest/public/search

[3] http://textblob.readthedocs.org/en/dev/

[4] https://github.com/dbpedia-spotlight/dbpedia-spotlight/wiki

[5] We observe that a substantial number of tweets contain a URL to a news article and the contents are just the news titles, which do not provide much additional information and opinions. We skip those cases and consider the tweets without URLs only.

4.3.2 Event and Aspect Discovery

The dynamic hierarchical entity-aware event/aspect discovery module proposed in the previous chapter is employed to learn the optimal representation of the news events and their multiple aspects. This module constructs an event-aspect hierarchy in a top-down manner recursively. The recursive function integrates text, entities and time with the intuition that an event/aspect must be coherent in all these dimensions.

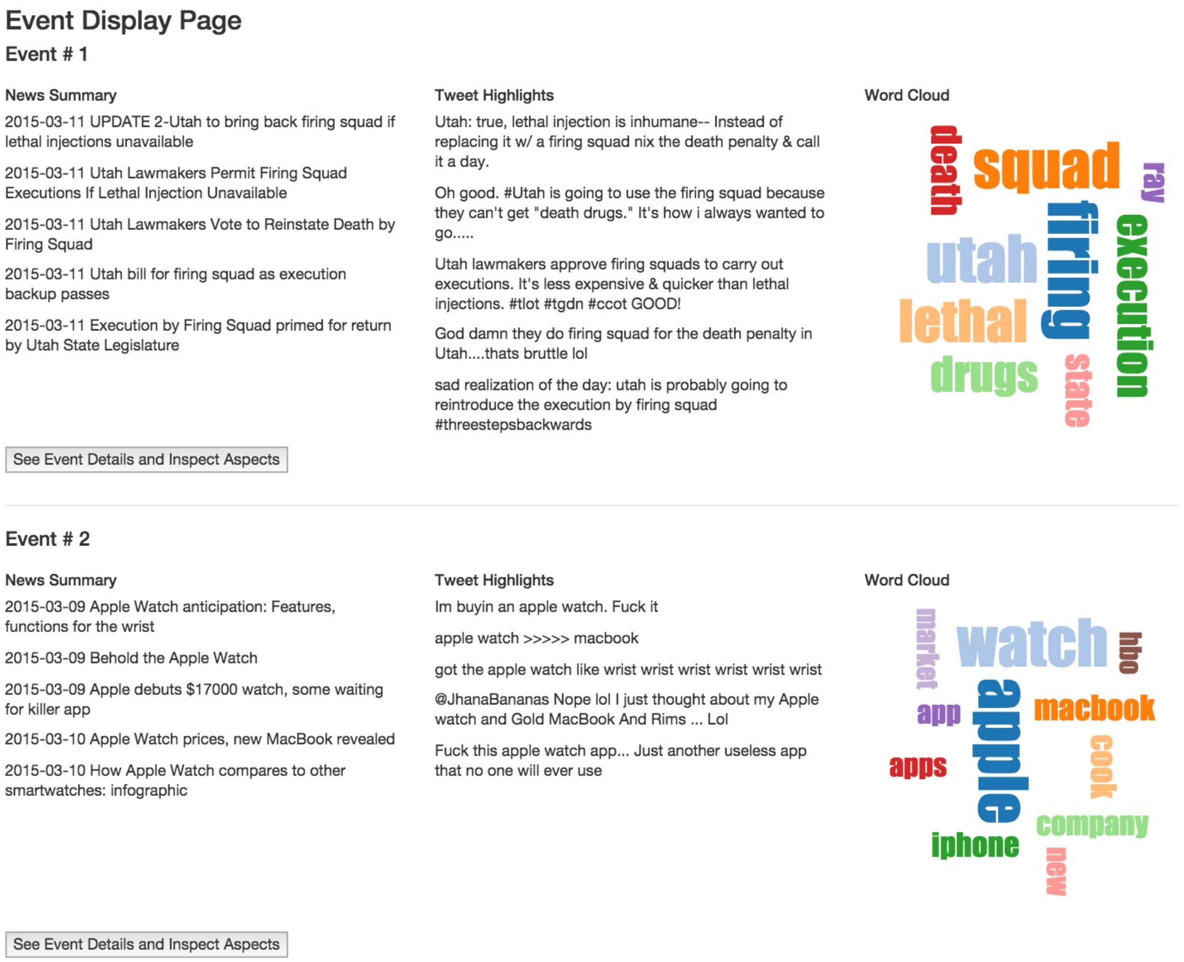
EKNOT first clusters the entire collection into events where each event z is described by a multinomial word distribution $\phi^{\mathtt{w}}$, a Gaussian time distribution $\mathcal{N}(\mu, \sigma)$ and multinomial entity distributions $\{\phi^{\mathtt{e}}\}, \mathtt{e} = \mathtt{p}, \mathtt{l}, \mathtt{o}$. A multinomial topic (event) distribution θ for each news article is also estimated to find aspects. Then for each event, the model decomposes the collection according to $\{\theta\}$ and applies the above procedure to obtain the descriptor of each aspect. The descriptor for an event/aspect is illustrated in the left box of Figure 4.4.

In our system, the event level computation is always performed once a user issues a query, while the aspect discovery is performed only if a user finds some event interesting and decides to investigate it.

4.3.3 Tweets Linking

Connecting a single news article to its relevant tweets is an active research area. The tweets linking module is ready to take advantage of any existing methods even though what we consider here is to connect an *event* to its relevant tweets.

The goal of this step is to maintain a high-quality news list and a high-quality tweet list for each event with an emphasis on relevance. The two lists will be used to generate summaries in the co-ranking component in Section 4.3.4 and to analyze opinions in Section 4.3.4. For the sake of both effectiveness and efficiency, we design the following procedure to select tweets for each event as illustrated in Figure 4.4. EKNOT first obtains a list of news articles for each event based on $\{\theta\}$, *i.e.*, a news article d with $P(z|d)$ greater than a threshold will



Event Display Page

Event # 1

News Summary

2015-03-11 UPDATE 2-Utah to bring back firing squad if lethal injections unavailable

2015-03-11 Utah Lawmakers Permit Firing Squad Executions If Lethal Injection Unavailable

2015-03-11 Utah Lawmakers Vote to Reinstate Death by Firing Squad

2015-03-11 Utah bill for firing squad as execution backup passes

2015-03-11 Execution by Firing Squad primed for return by Utah State Legislature

Tweet Highlights

Utah: true, lethal injection is inhumane-- Instead of replacing it w/ a firing squad nix the death penalty & call it a day.

Oh good. #Utah is going to use the firing squad because they can't get "death drugs." It's how i always wanted to go.....

Utah lawmakers approve firing squads to carry out executions. It's less expensive & quicker than lethal injections. #tlot #tgdn #ccot GOOD!

God damn they do firing squad for the death penalty in Utah....thats bruttle lol

sad realization of the day: utah is probably going to reintroduce the execution by firing squad #threestepsbackwards

Word Cloud

Event # 2

News Summary

2015-03-09 Apple Watch anticipation: Features, functions for the wrist

2015-03-09 Behold the Apple Watch

2015-03-09 Apple debuts $17000 watch, some waiting for killer app

2015-03-10 Apple Watch prices, new MacBook revealed

2015-03-10 How Apple Watch compares to other smartwatches: infographic

Tweet Highlights

Im buyin an apple watch. Fuck it

apple watch >>>>> macbook

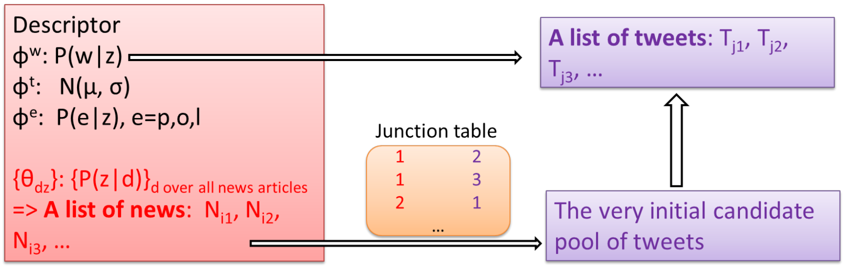
got the apple watch like wrist wrist wrist wrist wrist wrist

@JhanaBananas Nope lol I just thought about my Apple watch and Gold MacBook And Rims ... Lol

Fuck this apple watch app... Just another useless app that no one will ever use

Word Cloud

Figure 4.3: Event Display Page



Descriptor
ϕ^w: P(w|z)
ϕ^t: N(μ, σ)
ϕ^e: P(e|z), e=p,o,l
$\{\theta_{dz}\}$: $\{P(z|d)\}_{d \text{ over all news articles}}$
=> A list of news: N_{i1}, N_{i2}, N_{i3}, ...
Junction table
1 2
1 3
2 1
...
A list of tweets: T_{j1}, T_{j2}, T_{j3}, ...
The very initial candidate pool of tweets

Figure 4.4: Select the candidate tweets with an event descriptor. Output: a news list and a tweet list.

be selected for event z. The linked tweets (based on the junction table) of the selected news articles form the very initial pool of the candidate tweets. Then a list of tweets are obtained by standard information retrieval with BM25[6] using the event's word distribution $\phi^{\mathtt{w}}$ as a query. Up to now, for each event we have obtained a list of news and a list of relevant tweets.

Within each event, EKNOT uses the aspect descriptors to select relevant tweets from the event's list of tweets for each aspect. It obtains the news list and tweet list in the same way as it does for an event, except that the initial pool takes the event's tweet list rather than referring to the junction table.

4.3.4 Joint Summarization

To present an event to users in an informative way, EKNOT offers a content summary from news for an objective description, highlights from tweets for an opinion overview, an entity graph, a time span and an opinion distribution.

Co-Ranking News and Tweets to Obtain News Summaries and Tweet Highlights

Our goal of this co-ranking step is to construct an objective news summary and an opinion-rich tweet summary for each event/aspect using its news list and tweet list obtained in Section 4.3.3. EKNOT co-ranks the news and tweets considering a) content and temporal *consistency* with the event/aspect; b) *coherence* between the news summary and tweet summary; c) *coverage* and *diversity* of the news(tweet) summary; and d) whether the tweet summary contains substantial *opinions/sentiments* and represents a general trend of the public.

To instantiate the co-ranking algorithm, EKNOT combines Co-HITS [13] and the Maximal Marginal Relevance (MMR) principle [8]. At the beginning, four static score lists are computed.

- R_n^c, R_t^c: Co-HITS score for news and tweets, which captures content consistency and coherence. Co-HITS is run on the bipartite news-tweets graph, where the edge weight

[6] http://en.wikipedia.org/wiki/Okapi_BM25

is the cosine similarity between the word vectors of the connected news and tweet. We impose a regularization term to enforce that the score of a news/tweet must be consistent with the BM25 score with the event's word distribution $\phi^{\mathtt{w}}$.

- R_n^t, R_t^t: Temporal consistency score for news and tweets, which captures temporal consistency, is computed as the probability density of the timestamp of a news/tweet.
- R_t^s: Sentiment polarity for tweets, given by a classifier which will be explained in Section 4.3.4.
- R_t^p: Popularity score for tweets based on Twitter-specific features such as retweet number and favorites.

The final news ranking is given by a linear combination of R_n^c and R_n^t, while tweets ranking is given by a linear combination of $R_t^c, R_t^t, R_t^s, R_t^p$. To guarantee the content and temporal coverage as well as diversity of the summaries, we iteratively penalizes redundant news/tweets under MMR. At last, the headlines of the top ranked news are output as the news summary[7] and the contents of the top ranked tweets are output as the tweet highlights.

Entity Graphs

Eknot generates an entity graph for each event/aspect with the descriptor. An entity is denoted by a node and the correlation between two entities is denoted by an edge. In our visualization, the node size is proportional to $P(e|z)$, indicating how influential an entity is. The edge width is proportional to the co-occurrence number of two entities within the event, indicating how strong the two entities are correlated. Different colors in an event's entity graph indicate different aspects and are consistent with the colors throughout the visualization.

[7] The headline is often a very precise summary of of a news article so we use headlines in the news summary. In our system, the full text of news articles are also accessible by clicking the headlines.

Time Spans

EKNOT utilizes the Gaussian time distribution $\phi^{\mathtt{t}}$ in the descriptor to approximate the time span of each event/aspect.

Opinion Analysis

Opinion analysis provides the sentiment polarity feature for co-ranking and is used to calculate the positive/negative percentage of public opinions towards an event/aspect. Naturally, the opinions are obtained from tweets. We are most interested in subjective tweets defined to contain "a personal positive or negative feeling" [23]. Tweets only covering pure facts such as repeating news headlines are considered neutral. In order to effectively extract subjective tweets and identify their sentiments, as well as to ensure efficiency, we build a two-step classification model to determine a tweet's sentiment following [3]:

- *Step 1.* Subjectivity Classification. This classifier decides whether a tweet is subjective or neutral. Tweets classified as subjective will be passed to step 2.

- *Step 2.* Polarity Classification. This classifier determines whether a subjective tweet is positive or negative.

In both steps, EKNOT builds a binary logistic regression classifier using unigram features, linguistic features such as punctuations, and dictionary-based features derived from SentiWordNet[8]. The neutral set in the training data is formed by news titles. The positive and negative sets are obtained from tweets by inspecting the emoticons[43]. The classifier in Step 1 is trained on all the three sets considering both positive and negative sets as subjective. In Step 2, we trains only on the positive set and the negative set.

The sentiment polarity scores weighted by their tweets' relevance to an event/aspect are aggregated to compute the positive and negative percentage of public opinions. Note that

[8] http://sentiwordnet.isti.cnr.it/

in our final presentation, we omit the neutral tweets because they're dominant in Twitter streams.

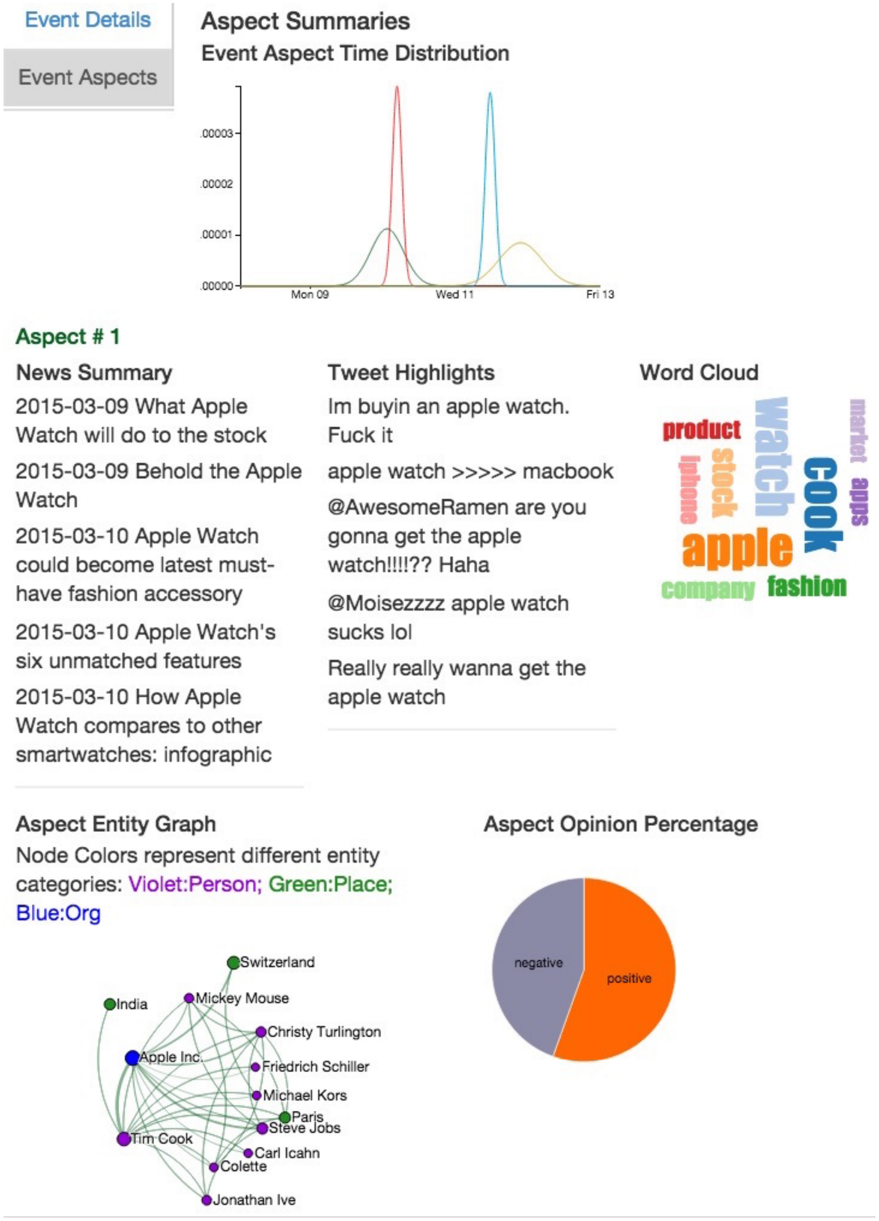
4.4 Demonstration Example

We start the demonstration by issuing a time period query. Here let us use `03/07/2015 - 03/14/2015` as an example. EKNOT returns the event display page which displays all the major events within this period, as illustrated in Figure 4.3. In addition to the news summary and tweet highlights, a word cloud is also presented. Entity graphs and opinion distributions are not displayed on this page to guarantee page loading efficiency as well as readability.

Users can choose any event to see the event details and inspect its aspects. By clicking on the button ("`See Event Details and Inspect Aspects`") below the summaries, users will be navigated to an "`event details`" page which displays the summary of the event[9]. To inspect the aspects of an event, users can click on the "`Event Aspects`" tab on the upper-left of this page. A list of aspects will be displayed as illustrated in Figure 4.5. This figure displays two sample aspects corresponding to Event # 2: one is about apple watch and the other about macbook. We are able to see the time spans on the top, news summaries on the left, tweet highlights on the right, following by an entity graph and a pie chart showing the sentiment distribution.

An additional functionality we will demonstrate is keyword search. Along with the time period query, users can also specify keywords to obtain only the events which match their information need.

[9] The layout of the event details is exactly the same as that of an aspect.



Event Details

Event Aspects

Aspect Summaries

Event Aspect Time Distribution

Aspect # 1

News Summary

2015-03-09 What Apple Watch will do to the stock

2015-03-09 Behold the Apple Watch

2015-03-10 Apple Watch could become latest must-have fashion accessory

2015-03-10 Apple Watch's six unmatched features

2015-03-10 How Apple Watch compares to other smartwatches: infographic

Tweet Highlights

Im buyin an apple watch. Fuck it

apple watch >>>>> macbook

@AwesomeRamen are you gonna get the apple watch!!!!?? Haha

@Moisezzzz apple watch sucks lol

Really really wanna get the apple watch

Word Cloud

Aspect Entity Graph

Node Colors represent different entity categories: Violet:Person; Green:Place; Blue:Org

Aspect Opinion Percentage

Figure 4.5: Aspect Details

Chapter 5

Leveraging Social Media to Customize Event Profiling in Traditional News Media

5.1 Overview

Given the same major events or breaking news, why did Mr. Jon Stewart attract a significantly larger audience than an ordinary CNN anchor, and become irreplaceable for "The Daily Show"? Besides blending humor with the news, the show catches *relevant* and *interesting* facts (e.g., the words from Republican celebrities who opposed Obamacare) from the large set of news articles, while the general narrative facts (e.g., the numbers and coverage of Obamacare) provided by regular television news may distract the listeners from the stories in politics. In this chapter, we focus on the practical problem of *customized news event profiling*, which aims at ranking the news sentences in a listener-centric way with both relevance and interestingness.

Incorporating interestingness into the news data is nontrivial: it is impossible to collect the reflection from the crowd before the event profile comes out. We consider to address this issue from a novel angle where social media information are leveraged to bridge the gap between the plain texts and the listeners' interests. Taking social impact into account, a customized event profile ranks news sentences and tweets in a way that not only captures relevant aspects of an event, but also reflects people's interests.

Leveraging knowledge from the social media is promising but also rather challenging. The news data and tweets cannot get connected by their sources: there is little overlap between the news proxies and users' Twitter accounts. The distinct language styles of news articles and social media posts further place difficulty on aligning them effectively. On the other hand, manually labeling the massive tweets as related or unrelated to the given news documents is not feasible. Therefore, we propose a novel unsupervised graph-based method to incorporate social impact into event profiles. We introduce a *"news-content unit-tweet"* tripartite graph (Figure 5.1) in which the news sentences and tweets are naturally connected via content units. Content units here are defined as natural and meaningful semantic units appearing in news articles and tweets. A propagation model which seamlessly combines global and local context is devised on this graph to effectively propagate social impact information from tweets to news.

A customized event profile consists of ranked lists of *content units*, *news sentences* and *tweets*. The ranking of news sentences should be influenced by tweets in a way that the highly ranked news sentences are more interesting to the users who posted those tweets. Such interestingness is measured by the popularity of the tweets. The event profiles can be readily used to generate summaries for events, and they are expected to better reflect people's interest. Furthermore, given different user groups, either by age, by gender, or by location, if we confine tweets to each group, the customized profiles will reflect the interest drift from one group to another group, which not only can benefit real-world applications such as personalized news recommendation, but also can be of great interest to social scientists.

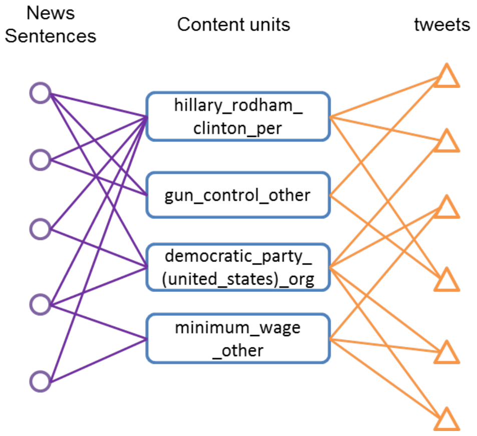


Figure 5.1: Bridging News and Tweets with Content Units

5.2 Event Discovery and Forming Global Context

Previous studies on event summarization often assume the existence of a set of relevant documents from which the summary is generated. Such sets of documents are usually handcrafted by human or generated by carefully designed queries. However, in practice, most of the time we are facing a collection of documents which contain all kinds of events without any annotation. Therefore, an unsupervised event discovery module is highly desirable to *automatically* identify the events and the corresponding documents (news articles and tweets), which serve as the input for subsequent tasks.

5.2.1 Event Discovery

Given a collection of news articles and a Twitter stream, our framework starts with automatic event discovery to identify a set of news articles and a set of tweets for each event. This step is achieved by generalizing the graphical model proposed in Chapter 3, which organizes topically coherent news articles into clusters (events) and then link relevant tweets to each cluster. The clustering step jointly makes use of text, entities and time jointly to enforce coherence through all these dimensions with a background topic absorbing common words/entities/timestamps. Instead of designating a distribution for each type of entities, here we introduce *content units* (formally defined in Definition 11) as a condensed representation for all types of entities. Multiple sets of entity distributions collapsed into one single distribution accordingly. This relaxation not only allows arbitrary types of entities, but also sheds light on which type of entities plays the most important role in a particular event.

The generative process is described as follows. The notations used throughout this chapter are summarized in Table 5.1. The plate notation for the generative model is given in Figure 5.2.

To generate each word in news article d,

Table 5.1: Summary of Notations

Symbol	**Description**
n, c, t	a news sentence, a content unit, and a tweet, respectively
A^N, A^T	$A^N = \{a^N_{nc}\}, A^T = \{a^T_{tc}\}$, the un-normalized affinity matrices. Both are symmetric.
$\phi^{\mathtt{w}}, \phi^{\mathtt{c}}$	the word distribution and the distribution over content units from the global context
$W^{NC}, W^{CN}, W^{TC}, W^{CT}$	$l1$ normalized transition matrices based on A^N and A^T by row (W^{NC}, W^{TC}) or by column (W^{CN}, W^{CT}) then transpose

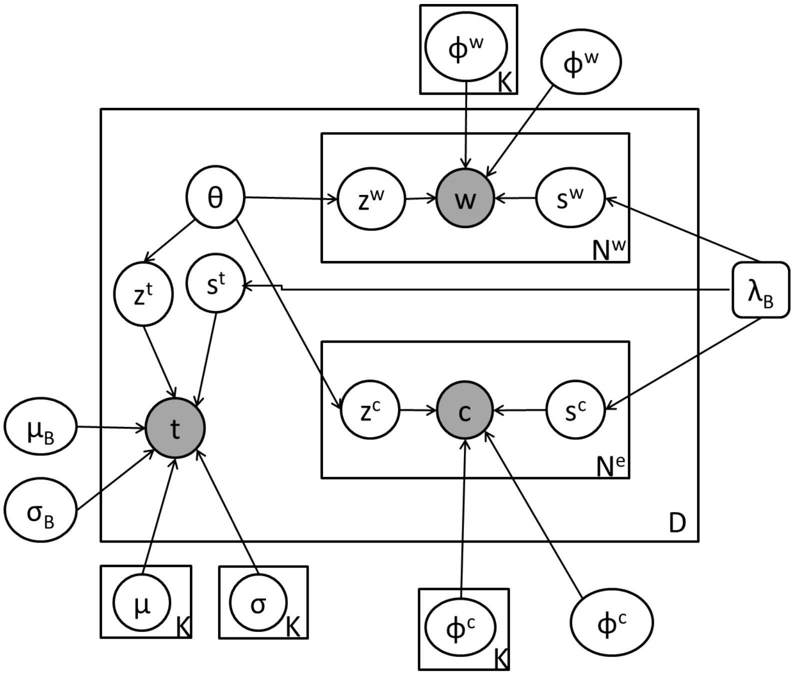


Figure 5.2: Plate Notation: Event Discovery

1. Draw a switch variable $s^{\mathtt{w}} \sim Bernoulli(\lambda_B)$. λ_B is the topic proportion of the background topic B. [1]

2. If $s^{\mathtt{w}} = 1$,

   draw a word w from the background topic B: $w \sim \phi^{\mathtt{w}}_B$;

   Else,

   draw a topic $z^{\mathtt{w}}$ from the topic distribution θ_d,

[1] The background topic B is specified by the entire collections word/content unit/time distributions. λ_B is a hyper parameter.

draw a word w from the topic $z^{\mathtt{w}}$: $w \sim \phi^{\mathtt{w}}_{z^{\mathtt{w}}}$.

To generate a timestamp t_d for news article d,

1. Draw a switch variable $s^{\mathtt{t}} \sim Bernoulli(\lambda_B)$.
3. If $s^{\mathtt{t}} = 1$,

   draw a timestamp t_d from the background time distribution B: $t_d \sim \mathcal{N}(\mu_B, \sigma_B)$;

   Else,

   draw a topic $z^{\mathtt{t}}$ from the topic distribution θ_d,

   draw a timestamp t_d from the topic $z^{\mathtt{t}}$: $t_d \sim \mathcal{N}(\mu_{z^{\mathtt{t}}}, \sigma_{z^{\mathtt{t}}})$.

To generate each content unit c in news article d,

1. Draw a switch variable $s^{\mathtt{c}} \sim Bernoulli(\lambda_B)$.
2. If $s^{\mathtt{c}} = 1$,

   draw a content unit c from the background topic B: $c \sim \phi^{\mathtt{c}}_B$;

   Else,

   draw a topic $z^{\mathtt{c}}$ from the topic distribution θ_d,

   draw an content unit c from the topic $z^{\mathtt{c}}$: $c \sim \phi^{\mathtt{c}}_{z^{\mathtt{c}}}$.

A news article is assigned to the most relevant event (topic) based on the topic distribution θ_d; and a tweet is linked to the most relevant event by the following language model:

$$\begin{aligned} &\log P(d|z) \quad (d \text{ is a tweet and } z \text{ is an event}) \\ =&\log P(d^{\mathtt{w}}|z) + \log P(d^{\mathtt{c}}|z) + \log P(d^{\mathtt{t}}|z) \end{aligned}$$

where $d^{\mathtt{w}}$ and $d^{\mathtt{c}}$ denote the words and the content units in d, and $d^{\mathtt{t}}$ is the timestamp of d.

5.2.2 Global Context

The event discovery step automatically identifies a group of news articles and tweets corresponding to each event. The learned topics $\{\phi_z^{\mathtt{w}}\}$ naturally serves as *global contexts*, *i.e.*, the word distribution $\phi_z^{\mathtt{w}}$ summarizes event z at the global level.

DEFINITION 10 (Event and Global Context)**.** *An event z contains a set of news articles and a set of tweets, together with a global context which is defined by a multinomial word distribution $\phi_z^{\mathtt{w}}$.*

5.3 Bridging News and Tweets by Content Units with Local Context

News articles and tweets are written in different styles by nature. It is critical to bridge the vocabulary gap between these two sources for joint analysis. Instead of directly matching news sentences and tweets via words, we propose to use *content units* as bridges for quality information transfer.

The formal definition of content units is given as follows.

DEFINITION 11 (Content Unit)**.** *Content units are key concepts that are natural and meaningful semantic units appearing in news articles and tweets. They are the indicators of the core content.*

In this paper, we use Wikipedia concepts (entities) as our content units for its broad coverage, high quality, timeliness, as well as its demonstrated robustness as semantic representations in the literature [16, 36]. The content units are extracted from both news articles and tweets using DBpedia Spotlight[2]. While we do not distinguish the entity types in the topic model, each content unit is aware of its type to facilitate data analysis. *Person*, *Place* and *Organization* are kept as they are and all other types are denoted as *Other*.

[2] https://github.com/dbpedia-spotlight/dbpedia-spotlight

5.3.1 Graph Construction

While different vocabularies may be used for news and tweets, especially descriptive words such as adjectives and adverbs, content units are less versatile, which can be good indicators of various contents. Therefore, we use content units as "anchors" to propagate social impact from tweets to news. As shown in Figure 5.1, a tripartite graph is constructed with three types of nodes: news sentences $\{n\}$, content units $\{c\}$ and tweets $\{t\}$.

Our tripartite graph contains two contexts: a news context N and a tweet context T. A bipartite graph is constructed out of each context. A context window with size l is applied to identify the local context of a content unit. For each content unit c, we aggregate the words from the context windows of every occurrence c to form a bag-of-words vector representation.

DEFINITION 12 (Local Context)**.** *The size-l local context of a content unit c with respect to a document collection D is denoted by ϕ_c^D, where l is the size of the context window, and $D \in \{N, T\}$ (news or tweets, respectively).*

With the vector representations of the content units $\{\phi_c^D\}$, the edge weights a_{nc} and a_{tc} can be computed as the cosine similarity between a content unit and a news sentence (ϕ_n), or tweet (ϕ_t):

$$a_{nc} = \cos(\phi_n, \phi_c^N), \quad a_{tc} = \cos(\phi_t, \phi_c^T)$$

where ϕ_n and ϕ_t are the bag-of-words representations of a news sentence and a tweet, respectively. We obtain two affinity matrices $A^N = \{a_{nc}^N\}$ and $A^T = \{a_{tc}^T\}$.

It is worth highlighting that the affinity score on each edge is computed using the local context distribution of the content unit in the corresponding context (news/tweets). This design allows the content units to adapt to each source, as well as act as anchors to bridge the two sources.

5.3.2 Leveraging Local and Global Context to Customize Event Profiling

An event profile is defined as follows.

DEFINITION 13 (Event Profile)**.** *An event profile consists of a set of news sentences* $\{n\}$ *with scores* $\{s(n)\}$*, a set of tweets* $\{t\}$ *with scores* $\{s(t)\}$*, and a set of content units* $\{c\}$ *with two set of scores* $\{s^N(c)\}$ *and* $\{s^T(c)\}$*, corresponding to the importance in news and tweets, respectively.*

The news sentences and tweets are identified for each event as described in Section 5.2. Our goal is to learn the scores of the news sentences and tweets. The following properties are desired for a good event profile: (a) Top ranked news sentences and tweets in the profile should be consistent with the global context of the event; (b) Top ranked news sentences and tweets should reflect the social impact. (c) Top ranked news sentences and tweets should be coherent with each other.

To accommodate the above properties, we propose a novel graph based method seamlessly combining global and local context. The local context is encoded in the representation of the content units and is used to compute the strength of a link between a content unit and a sentence/tweet, which directly affects the graph structure. The global context acts as a regularization to the scores of news sentences and tweets. The global scores of news sentences and tweets in an event z are computed as follows:[3]

$$s_0(n) = \cos(\phi_n, \phi_z^{\mathtt{w}}), s_0(t) = \cos(\phi_t, \phi_z^{\mathtt{w}})$$

In order to impose *social impact* onto the graph, we boost the edge weight $\{a_{tc}\}$ by a multiplier of the popularity of a tweet t for all cs. We vectorize all the tweets using the bag-of-words representation and convert the vectors to be binary. These binary vectors are

[3] The global scores are normalized such that $\sum_n s_0(n) = 1$ and $\sum_t s_0(t) = 1$.

used as the signatures of tweets. After removing rare words, user names, 'RT's, and URLs, tweets with the same signature are grouped together. Each group is assigned one tweet node and the size of the group is used as the popularity multiplier.

5.3.3 Propagation Model

Figure 5.3 illustrates the propagation path of our model. The scores are first propagated from news sentences/tweets to content units. A content unit further disseminates its score to the other side, where mutual reinforcement takes place to co-rank the sentences and tweets.

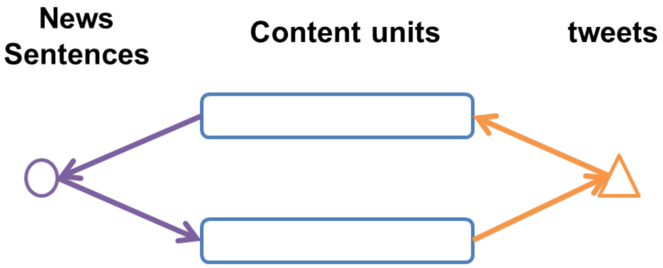


Figure 5.3: Propagation Path

The transition matrices W^{NC}, W^{TC} are obtained by normalizing the affinity matrices A^N and A^T by column; and W^{CN}, W^{CT} by normalizing A^N and A^T by row and then taking the transpose. Algorithm 2 sketches the proposed propagation model. The proof of convergence is given as follows.

Proof of Convergence for Algorithm 2. We show that the iterative updates are guaranteed to converge under mild assumptions on the transition matrices $W^{NC}, W^{TC}, W^{CN}, W^{CT}$. We further define $W^{NT} = W^{NC}W^{CT}$ and $W^{TN} = W^{TC}W^{CN}$. Due to the fact that $W^{NC}, W^{TC}, W^{CN}, W^{CT}$ are transition matrices, we have W^{NT}, W^{TN} as the aggregated transition matrices from news to tweets and tweets to news, respectively. If we only consider W^{NT} and W^{TN}, the triple graph is reduced into a bipartite graph of tweets and news. Hence,

```
Algorithm 2: Propagation Model
Input: A tripartite graph with news sentences $\{n\}$, content units $\{c\}$ and tweets $\{t\}$.
       Scores from global context $s_0(n)$, $s_0(t)$. The parameters balancing the global
       context and local context: $\lambda^N$ and $\lambda^T$. Number of iterations $Maxiter$.
Output: The scores for news sentences $\{s(n)\}$, tweets $\{s(t)\}$ and content units
        $\{s^N(c)\}$ and $\{s^T(c)\}$.
Initialize $s(n)$ with $s_0(n)$.
for $iter$ = 1:Maxiter do
    Information transfer from news to tweets:
        $s^N(c) = \sum_{n \in \mathcal{N}} w_{nc}^{NC} s(n)$    (5.1)
        $s(t) = (1 - \lambda^T) s_0(t) + \lambda^T \sum_{c \in \mathcal{C}} w_{ct}^{CT} s^N(c)$    (5.2)
    Information transfer from tweets to news:
        $s^T(c) = \sum_{t \in \mathcal{T}} w_{tc}^{TC} s(t)$    (5.3)
        $s(n) = (1 - \lambda^N) s_0(n) + \lambda^N \sum_{c \in \mathcal{C}} w_{cn}^{CN} s^T(c)$    (5.4)
end
```

with $k = 1, 2, ...$ as the update iterator, the iterative updates can be simplified as

$$\begin{aligned} s_t^k &= (1 - \lambda^T) s_t^0 + \lambda^T \sum_{c \in \mathcal{C}} w_{ct}^{CT} \sum_{n \in \mathcal{N}} w_{nc}^{NC} s_n^k \\ &= (1 - \lambda^T) s_t^0 + \lambda^T \sum_{n \in \mathcal{N}} w_{nt}^{NT} s_n^k; \end{aligned} \tag{5.5}$$

$$\begin{aligned} s_n^k &= (1 - \lambda^N) s_n^0 + \lambda^N \sum_{c \in \mathcal{C}} w_{cn}^{CN} \sum_{t \in \mathcal{T}} w_{tc}^{TC} s_t^{k-1} \\ &= (1 - \lambda^N) s_n^0 + \lambda^N \sum_{t \in \mathcal{T}} w_{tn}^{TN} s_t^{k-1}, \end{aligned} \tag{5.6}$$

where s_n^k and s_t^k are the scores of news n and tweet t at update step k, and $s_t^0 = s_0(t), s_n^0 = s_0(n)$.

In what to follow, we show that the iterative updates of (5.5) and (5.6) are guaranteed to converge under mild assumptions on the transition matrices W^{NT}, W^{TN}. Illustratively, we analyze (5.5). Same analysis can be straightforwardly applied to (5.6).

Substituting (5.6) into (5.5), we have

$$\begin{aligned} s_t^k &= (1-\lambda^T)s_t^0 + \lambda^T(1-\lambda^N)\sum_{n\in\mathcal{N}} w_{nt}^{NT} s_n^0 \\ &\quad + \lambda^T\lambda^N \sum_{n\in\mathcal{N}} w_{nt}^{NT} \sum_{t'\in\mathcal{T}} w_{t'n}^{TN} s_{t'}^{k-1} \\ &= (1-\lambda^T)s_t^0 + \lambda^T(1-\lambda^N)\sum_{n\in\mathcal{N}} w_{nt}^{NT} s_n^0 \\ &\quad + \lambda^T\lambda^N \sum_{t'\in\mathcal{T}} w_{t't}^{TT} s_{t'}^{k-1} \\ &= (1-\lambda^T\lambda^N)p_t + \lambda^T\lambda^N \sum_{t'\in\mathcal{T}} w_{t't}^{TT} s_{t'}^{k-1}, \end{aligned} \tag{5.7}$$

where $w_{t't}^{TT} = \sum_{n\in\mathcal{N}} w_{nt}^{NT} w_{t'n}^{TN}$ and

$$p_t = \frac{(1-\lambda^T)s_t^0 + \lambda^T(1-\lambda^N)\sum_{n\in\mathcal{N}} w_{nt}^{NT} s_n^0}{1-\lambda^T\lambda^N}.$$

In order to prove the convergence of (5.5) and (5.6), we can show that (5.7) is equivalent to the update of random walk on the tweet-tweet network. Since W^{TT} is also a transition matrix, what remains to prove is that $\sum_{t\in\mathcal{T}} p_t = 1$.

We first establish the following equality that

$$\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}} w_{nt}^{NT} s_n^0 = \sum_{n\in\mathcal{N}} s_n^0 \sum_{t\in\mathcal{T}} w_{nt}^{NT} = \sum_{n\in\mathcal{N}} s_n^0 = 1, \tag{5.8}$$

where the second equality is due to the fact that W^{NT} is a transition matrix and the last equality follows from $\sum_{n\in\mathcal{N}} s_n^0 = 1$.

Thus we have

$$\begin{aligned}
&\sum_{t\in\mathcal{T}}(1-\lambda^T)s_t^0 + \lambda^T(1-\lambda^N)\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}} w_{nt}^{NT} s_n^0 \\
&= (1-\lambda^T) + \lambda^T(1-\lambda^N) \\
&= 1-\lambda^T\lambda^N, \qquad (5.9)
\end{aligned}$$

where the first equality follows from $\sum_{t\in\mathcal{T}} s_t^0 = 1$ and (5.8). It follows immediately by (5.9) that $\sum_{t\in\mathcal{T}} p_t = 1$.

We have shown that (5.5) can be reduced to (5.7), which is equivalent to random walk on a tweet graph with transition matrix W^{TT}. Thus, (5.7) is guaranteed to converge if W^{TT} does not have two same dominate eigenvalues. Consequently, (5.5) and (5.6) converge. □

5.4 Empirical Study

Evaluating customized event profiles is a difficult task. The main difficulty comes from the impossibility of building a fair gold-standard against which the results can be compared. It is hard to determine what a "correct" customized profile is due to the subjective nature of our problem, as well as the infeasibility of requiring annotators to read the entire Twitter stream to come up with the most "interesting" sentences, making standard Rouge measures used in traditional summarization tasks hardly applicable.

In this section, we try to quantify the customization by performing comparative analysis of our customized event profiles against the results given by LexRank[15], which is used as a base measure. Our hypothesis is that the scores given by running LexRank on news articles represent an *unbiased* ranking of the news sentences; the deviation of the customized ranking from LexRank is caused by introducing tweets. We investigate the drift of content units, words, and sentences.

We first present a case study with extensive data analysis, aiming to validate that our method can generate high quality customized news event profiles if the interest drift does exist. Then we report results from a carefully designed user study which offers more evidence on the effectiveness of our method, as well as providing insights into how significant users' interest in social media drift from mainstream news media. At the end, we describe an interesting application on the proposed framework: interest-driven news event profiling by "partial content unit activation".

5.4.1 Data Collection

We keep an automatic crawler[4] running which scrapes the top stories from Google News[5] every 30 minutes. For each news article, the crawler queries Twitter Search API[6] with extracted noun phrases from the title and snippet of the news article. Tweets containing at least two of the noun phrases are returned. We collected tweets that are posted within one day after the published time of the news article. The experiment dataset we used in this study contains a week of data between Feb 07, 2016 and Feb 13, 2016. The dataset consists of $1,012$ news articles and $1,761,447$ tweets in total.

5.4.2 Implementation Details

The number of iterations in the topic model is set to 20. The topic modeling parameters are initialized by the results from k-means clustering with 50 random initializations. The number of topics (events) is empirically set to 70. 5 out of the 70 events are chosen to form a development dataset to tune the propagation model. We found that the results are not sensitive to the number of content units, or λ^N and λ^T when they are greater than 0.5.[7] In what follows, top 100 content units are included in the propagation model according to ϕ_z^c

[4] This is the same crawler as we used in Chapter 3.

[5] https://news.google.com/

[6] https://dev.twitter.com/rest/public/search

[7] Larger λ^N and λ^T allow more impact from the propagation rather than the global scores.

(See Section 5.2) in the corresponding event z. The iterations stop at a tolerance of $1e$-9 using $l2$ norm. $\lambda^N = \lambda^T = 0.7$. For all the bag-of-words representations in our model, tf-idf weighting is applied.

5.4.3 A Case Study on "2016 Taiwan Earthquake"

An earthquake with a moment magnitude of 6.4 struck Pingtung City in southern Taiwan, having a maximum intensity of VII (Very strong) on the Mercalli intensity scale, causing widespread damage and 117 deaths. Almost all of the deaths were caused by a collapsed apartment building. The earthquake was the deadliest earthquake in Taiwan since the 921 earthquake in 1999.[8] This event is manually identified from the output of the topic model. There are a total of 282 news sentences and 8466 tweets in this event.

Drift of Content Units and Words

We investigate the scores of content units inbound from news: $\{s^N(c)\}$. For LexRank, we compute $s^N(c)$ according to Eq. 5.1 but substitute the scores of news sentences $\{s(n)\}$ with the scores output by LexRank. The news scores are normalized to sum to 1 for both methods so that the scores of content units form a distribution. Plotting these two distributions with the horizontal axis being ordered according to LexRank scores, we observe the drift between the distributions as shown in Figure 5.4(a). Similarly, we plot the word distributions based on

$$s^N(w) = \sum_{n \in \mathcal{N}} p(w|n)s(n)$$

where $p(w|n) = \text{tf-idf}(w)/\sum_{w \in n} \text{tf-idf}(w)$, as shown in Figure5.4(b).

While the two distributions follow the same trend in general, the customized profile gives spikes to certain content units/words. If we look into the spikes, the content units and words which generate these spikes can be obtained. We investigate the top 20 content units and

[8] https://en.wikipedia.org/wiki/2016_Taiwan_earthquake

words given by $\phi_z^{\mathtt{c}}$ and $\phi_z^{\mathtt{w}}$, and print the content units/words which have higher probability mass than that in LexRank, *i.e.*, $s^N_{customized}(c)/s^N_{LexRank}(c) > 1$, as shown in Figure 5.5 and 5.6. To complement the analysis, we also print the content units/words whose position move up more than 4. The content units and words may not seem to make a lot of sense now but they will become clear as we investigate the drift of sentences.

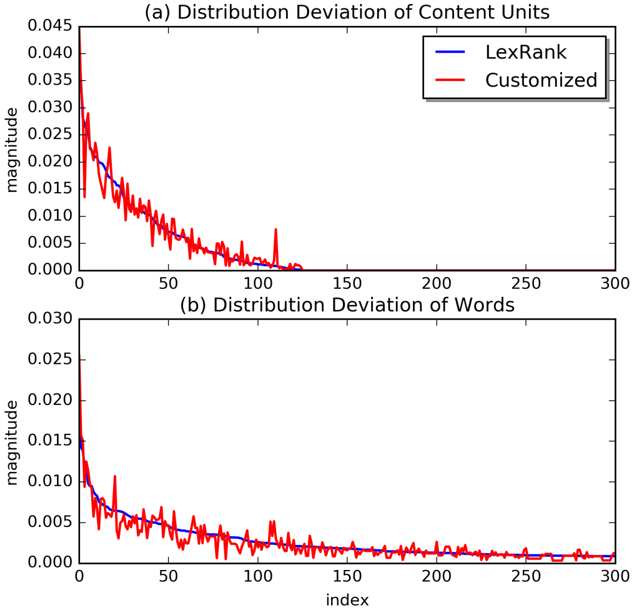
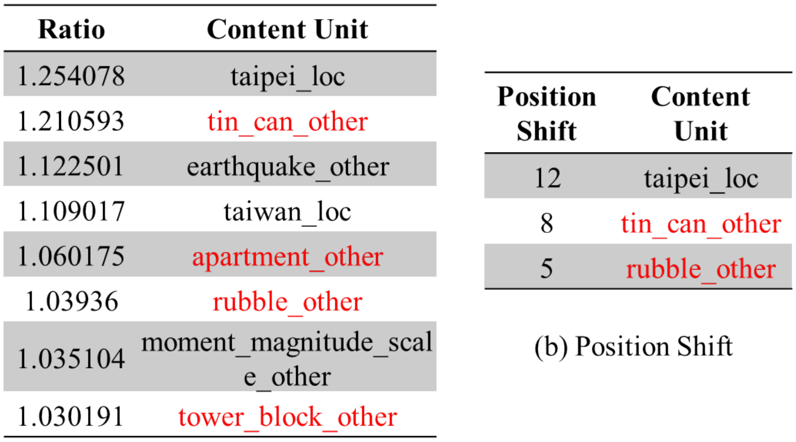


Figure 5.4: The Distributions of Content Units and Words. Spikes indicate the deviation of the customized distribution from LexRank.

Drift of Sentences

In order to investigate the ranking difference of sentences, we generate a length-K (K sentences) summary for both models. K is empirically set to $log(\#sentences)$. The summary is generated by adding sentences in rank order, but discards any sentences that are too similar to the ones already placed in the summary to remove redundancy based on Cross-Sentence Information Subsumption (CSIS) [42].



Ratio	Content Unit
1.254078	taipei_loc
1.210593	tin_can_other
1.122501	earthquake_other
1.109017	taiwan_loc
1.060175	apartment_other
1.03936	rubble_other
1.035104	moment_magnitude_scale_other
1.030191	tower_block_other

(a) Ratio

Position Shift	Content Unit
12	taipei_loc
8	tin_can_other
5	rubble_other

(b) Position Shift

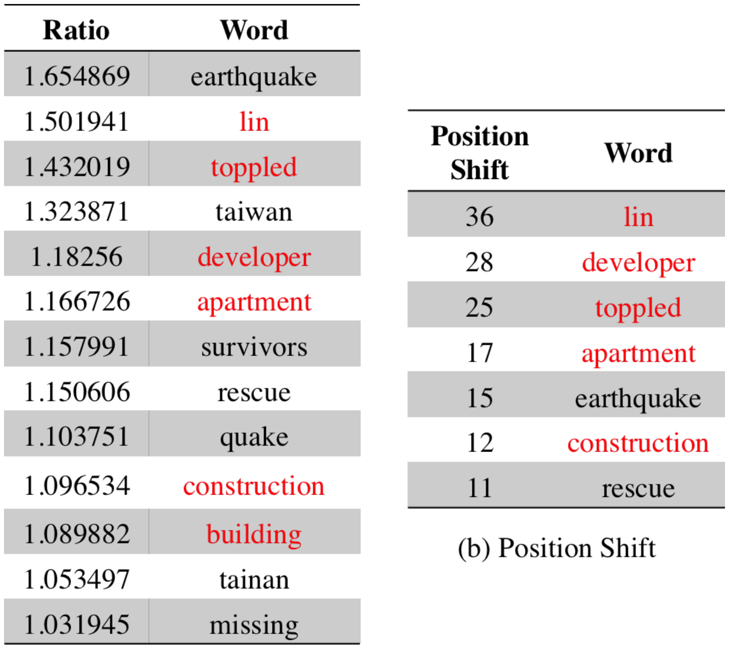
Figure 5.5: Deviation of Content Units

The summaries generated are shown in Table 5.2. Together with the news summary, we also generate a length-K[9] tweet summary from our customized event profile. The popularity of each tweet is printed at the start of each line.

The summary from LexRank focused on the objective facts of the earthquake, where it reports casualties, rescue efforts, and presents facts of historical earthquakes. However, if we look at the top ranked tweets, while the rescue efforts did attract much attention, people paid even more attention to the collapsed apartment building. Tin cans built into the walls of the toppled complex raised people's concern. The arrest of the devolopers of the collapsed building also got substantially tweeted. Now let's look at the customized summary. Asides from the key casualty facts and rescue efforts, discussions on the cause of the collapsed building, as well as the arrest of the building devolopers are also presented, which correctly reflects the social impact.

Back to the deviations of content units and words, we can observe that the content units/words highlighted in red accurately capture the public's attention on the collapsed building. Most of them are self-explanatory now. The word "lin" is the family name of the chairman of the developer company that built the collapsed building.

[9] $K = log(\#tweets)$



Ratio	Word
1.654869	earthquake
1.501941	lin
1.432019	toppled
1.323871	taiwan
1.18256	developer
1.166726	apartment
1.157991	survivors
1.150606	rescue
1.103751	quake
1.096534	construction
1.089882	building
1.053497	tainan
1.031945	missing

(a) Ratio

Position Shift	Word
36	lin
28	developer
25	toppled
17	apartment
15	earthquake
12	construction
11	rescue

(b) Position Shift

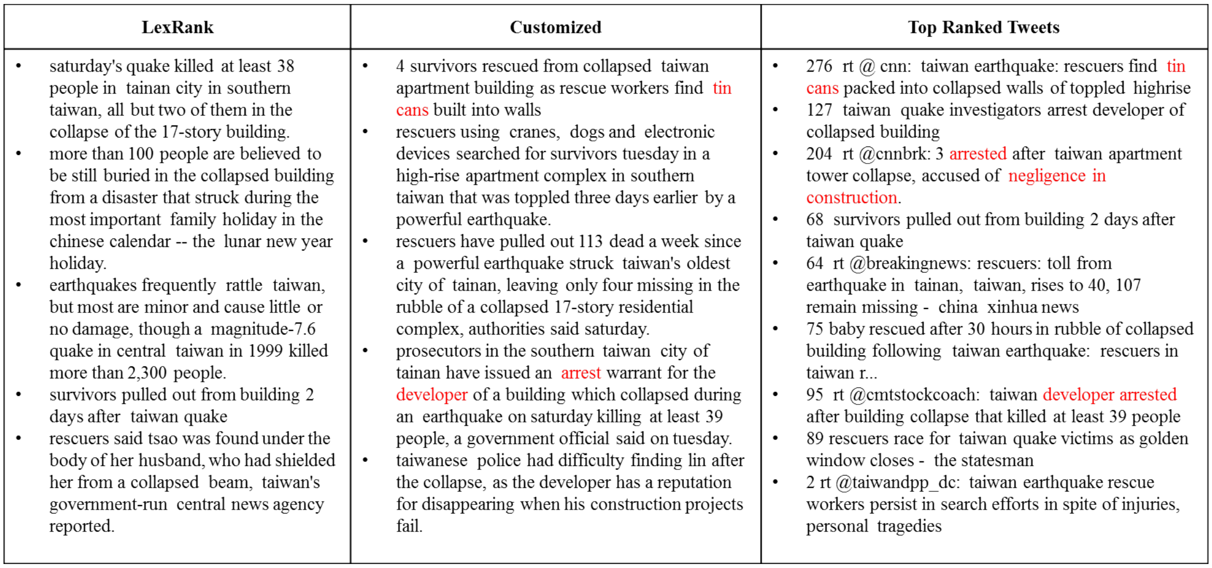
Figure 5.6: Deviation of Words

It is also worth noting that the tweets with higher popularity are ranked higher in general. This echos with our treatment of boosting the edge weights in $\{a_{tc}\}$ by the popularity of tweet t. However, we note that the rank of tweets are not solely determined by popularity. The content relevance also affects the ranking: popular but irrelevant tweets are demoted.

5.4.4 User Study

In order to obtain a more accurate measure of the summary quality, we performed a simple user study. For each event, a user was given four summaries: a) a tweet summary containing the most popular tweets, b) a news summary generated by LexRank, c) a news summary generated from the customized event profile, and d) a tweet summary generated from the customized event profile. The same CSIS post-processing was applied to all four systems to select sentences/tweets into summaries. Users were presented with four questions: (1) Overall quality: is each of the four summaries a good summary by itself in terms of informativeness and coherence? (2) Tweet coherence: is summary a) reasonably covered in

Table 5.2: Deviation of Sentences



LexRank	Customized	Top Ranked Tweets
• saturday's quake killed at least 38 people in tainan city in southern taiwan, all but two of them in the collapse of the 17-story building. • more than 100 people are believed to be still buried in the collapsed building from a disaster that struck during the most important family holiday in the chinese calendar -- the lunar new year holiday. • earthquakes frequently rattle taiwan, but most are minor and cause little or no damage, though a magnitude-7.6 quake in central taiwan in 1999 killed more than 2,300 people. • survivors pulled out from building 2 days after taiwan quake • rescuers said tsao was found under the body of her husband, who had shielded her from a collapsed beam, taiwan's government-run central news agency reported.	• 4 survivors rescued from collapsed taiwan apartment building as rescue workers find tin cans built into walls • rescuers using cranes, dogs and electronic devices searched for survivors tuesday in a high-rise apartment complex in southern taiwan that was toppled three days earlier by a powerful earthquake. • rescuers have pulled out 113 dead a week since a powerful earthquake struck taiwan's oldest city of tainan, leaving only four missing in the rubble of a collapsed 17-story residential complex, authorities said saturday. • prosecutors in the southern taiwan city of tainan have issued an arrest warrant for the developer of a building which collapsed during an earthquake on saturday killing at least 39 people, a government official said on tuesday. • taiwanese police had difficulty finding lin after the collapse, as the developer has a reputation for disappearing when his construction projects fail.	• 276 rt @ cnn: taiwan earthquake: rescuers find tin cans packed into collapsed walls of toppled highrise • 127 taiwan quake investigators arrest developer of collapsed building • 204 rt @cnnbrk: 3 arrested after taiwan apartment tower collapse, accused of negligence in construction. • 68 survivors pulled out from building 2 days after taiwan quake • 64 rt @breakingnews: rescuers: toll from earthquake in tainan, taiwan, rises to 40, 107 remain missing - china xinhua news • 75 baby rescued after 30 hours in rubble of collapsed building following taiwan earthquake: rescuers in taiwan r... • 95 rt @cmtstockcoach: taiwan developer arrested after building collapse that killed at least 39 people • 89 rescuers race for taiwan quake victims as golden window closes - the statesman • 2 rt @taiwandpp_dc: taiwan earthquake rescue workers persist in search efforts in spite of injuries, personal tragedies

d)? (3) Interest drift: does summary d) contain noticeable interest drift from b)? (4) Interestingness: if yes to (3), does summary c) better reflect user interest than b) in terms of coherence with d)?[10] The "biggest" 50 events (measured by the number of documents) are selected for user study. The study had 10 users and each was asked to annotate 10 summary tuples. Each tuple is annotated by two users. We remove 2 of the tuples because they got negative answers for all summaries in question (1). For the remaining 48 events, results are shown in Table 5.3. Cohen's κ [31] which measures inter-rater agreement is reported for each question. Despite the difficulty of the annotation task, we observe moderate to substantial agreements according to the guidelines given in Landis and Koch [31]. The four summaries are generally considered as good summaries and our tweet summary is also able to capture the most popular tweets. But the tweet summary generated by simply using the most popular ones are less robust than others. Interest drift is observed in 61.5% of the events, which indicates that more than half of the news articles can get a potentially more interesting summary. This further justifies the motivation of our work. For 73.7% of the

[10] Here d) is a better reference summary than a) because it also takes content relevance into account besides popularity.

time when interest drift exists, our customized summary is able to capture the interesting pieces of information. Thus we conclude that our model can effectively customize news event by leveraging tweets.

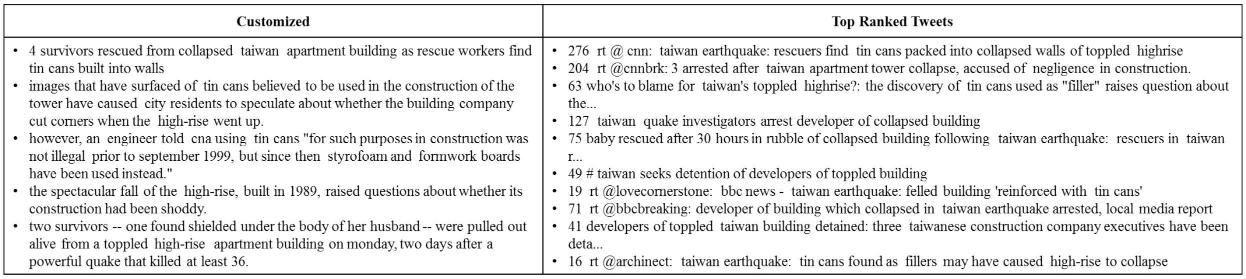
Table 5.3: Results from Manual Evaluation. The denominator for calculating the positive rate is the number of total agreements. The maximum possible number of total agreements for the first six questions is 48. For *interestingness*, we only consider the 24 tuples where both judgements indicate the existence of *interest drift*.

	Overall (a)	**Overall (b)**	**Overall (c)**	**Overall (d)**	**Tweet Coherence**	**Interest Drift**	**Interestingness**
% Positive	77.5%	93.2%	90.9%	87.8%	86.4%	61.5%	73.7%
#Total Agreements	40	44	44	41	44	39	19
Cohen's κ	0.578	0.562	0.619	0.5	0.7	0.612	0.524

5.4.5 Interest-Driven Summarization by *Partial Content Unit Activation*

With a ranked list of the content units, users can choose to activate part of them for personalized summaries, which ranks sentences and tweets based on a subset of the content units and encourages those closely relevant to the activated content units to rank at the top. Table 5.4 shows the results by activating only *tin_can_other* and *tower_block_other* in the *2016 Taiwan Earthquake* event. The resulting summary becomes concentrated on the discussions surrounding people's concerns about the use of tin cans in construction.

Table 5.4: Summary from Partial Activation of Content Units: *tin_can_other* and *tower_block_other*



Customized	**Top Ranked Tweets**
• 4 survivors rescued from collapsed taiwan apartment building as rescue workers find tin cans built into walls • images that have surfaced of tin cans believed to be used in the construction of the tower have caused city residents to speculate about whether the building company cut corners when the high-rise went up. • however, an engineer told cna using tin cans "for such purposes in construction was not illegal prior to september 1999, but since then styrofoam and formwork boards have been used instead." • the spectacular fall of the high-rise, built in 1989, raised questions about whether its construction had been shoddy. • two survivors -- one found shielded under the body of her husband -- were pulled out alive from a toppled high-rise apartment building on monday, two days after a powerful quake that killed at least 36.	• 276 rt @ cnn: taiwan earthquake: rescuers find tin cans packed into collapsed walls of toppled highrise • 204 rt @cnnbrk: 3 arrested after taiwan apartment tower collapse, accused of negligence in construction. • 63 who's to blame for taiwan's toppled highrise?: the discovery of tin cans used as "filler" raises question about the... • 127 taiwan quake investigators arrest developer of collapsed building • 75 baby rescued after 30 hours in rubble of collapsed building following taiwan earthquake: rescuers in taiwan r... • 49 # taiwan seeks detention of developers of toppled building • 19 rt @lovecornerstone: bbc news - taiwan earthquake: felled building 'reinforced with tin cans' • 71 rt @bbcbreaking: developer of building which collapsed in taiwan earthquake arrested, local media report • 41 developers of toppled taiwan building detained: three taiwanese construction company executives have been deta... • 16 rt @archinect: taiwan earthquake: tin cans found as fillers may have caused high-rise to collapse

5.5 Related Work

To the best of our knowledge, this is the first work leveraging social media to infuse news event profiling with people's interests. Yet it is related to existing studies of news media and social media.

5.5.1 Event Discovery/Summarization in Twitter

Numerous research efforts have been aimed at event discovery/summarization in Twitter [2, 44, 45, 46, 52], where various clustering methods taking well-calibrated features have been proposed. These studies focused on the single collection of tweets where huge number of random posts irrelevant to any news events interfere as noise.

5.5.2 Comparative Study of News and Tweets

Zhao et al. [66] conducted a comparative study on topic categories (politics/sports/etc.) and types (event-oriented/long-standing/etc.) of topics discovered from news and tweets by running separate topic models on the two sources. Subavsic and Berendt [48] performed a case study to investigate text/headline/sentiment/entity divergence between news and tweets in an aggregate sense. These studies extract statistics from each individual source separately and investigate the distribution differences. Being aware of the existence of the deviation of tweets from news, we take one step further to customize news event profiling with tweets. Our event profile ranks new sentences in a way that best accords with their social impact.

5.5.3 Joint Study of News and Tweets

Gao et al. [22] extracted complementary news sentences and tweets based on a joint topic model of news and tweets. Wei and Gao [59], Wei et al. [61], Yulianti et al. [65] studied news summarization utilizing linked tweets. Joint features of news and tweets turned out to significantly benifit this *single document* summarization task. A later work from Wei and Gao [60] explores effective ways using the tweets linked to news for generating extractive summary of each document. They reveal the very basic value of tweets that can be utilized by regarding every tweet as a vote for candidate sentences. They proposed unsupervised summarization models which leverage the linked tweets to master the ranking of candidate extracts via random walk. Compared to truly supervised summarizer unaware of tweets, this method achieves significantly better results with reasonably small tradeoff on latency, which motivates us to study news and tweets in a completely unsupervised setting.

Our work distinguishes itself from the above studies in two aspects: 1) The joint study is performed on *event* level, which not only can be readily used for *multi-document* summarization, but also captures the interest drift for an *event* from news media to social media; and 2) The existing work all target at recovering a gold standard summary generated solely from news, aiming to justify that additional information from Twitter can improve the quality of news summaries. In sharp contrast, our work intends to capture the *drift* from the "canonical" news summary by involving tweets. As noticed in the experimental study, such drift exists in more than half of the events. The event summaries generated by our model may deviate from the news gold standard but better accord with people's interest.

5.6 Discussions and Future Work

We discuss the limitations of our framework in this section. We point out the components where potential improvements can be made and provide alternative designs for future work.

5.6.1 Content Units

Content units play an important role in our method. In this exploratory study, Wikipedia entities are used for its broad coverage, reasonable timeliness, and high quality demonstrated in the literature, as well as simplicity. However, the entity structure in the current setting is flat. In reality, the semantic meanings of two content units may be overlapping, or at different granularity. Allowing ontology within the content units may capture even more accurate information. Another limitation of using Wikipedia entities is the latency. Newly emerging entities may not get updated as soon as they attract people's attention. Therefore, identifying new entities in an online fashion will make our application more practical for real-time users.

5.6.2 Graph Construction

The edge weights in our graph are computed based on local contexts. In this paper, the vector representation of a local context takes the bag-of-words representation and the affinity matrices are computed based on cosine similarity. However, it is worth exploring the latest sentence embedding techniques inspired by the success of word2vec [38] to see whether there is a more optimized vector representation. Other similarity measures would also be interesting to investigate.

5.7 Summary

In this chapter, we have presented an exploratory study of leveraging tweets to customize news event profiling. A propagation model simultaneously exploring global and local context was developed on a tripartite graph where news sentences and tweets are bridged by content units. We demonstrate that leveraging tweets can generate more interesting news summaries by extensive data analysis on a case study as well as manual evaluation.

Chapter 6

Conclusions

I have addressed two major research problems in mining social media data: 1) How can we systematically model heterogeneous data dimensions to model multidimensional user preference in social media? and 2) How can we effectively integrate external sources to achieve quality knowledge discovery from massive noisy social media data.

This dissertation first presents a general discriminative learning approach [56] for modeling multi-dimensional knowledge in a *supervised setting*. A learning protocol is established to model both explicit and implicit knowledge in a principled manner, which applies to general classification/prediction tasks. This approach accommodates heterogeneous data dimensions with a significant boosted expressiveness of existing discriminative learning approaches. It stands out with its capability to model *latent features*, for which arbitrary generative assumptions are allowed. A concrete instantiation of this model is given in the application of modeling users' time varying check-in preference in social media platforms. The prediction accuracy is significantly improved over the state-of-art models.

Social media data are unstructured, fragmented and noisy. In addition, most real applications come with no available annotation in an *unsupervised setting*. This dissertation addresses these challenges from a novel angle where external sources such as news media and knowledge bases are exploited to provide supervision. A unified framework is developed which links traditional news data to Twitter and enables effective knowledge discovery such as event detection and summarization [32, 55]. This framework complements the aforementioned discriminative approach to model multidimensional knowledge taking a generative learning approach.

Along the line of integrated news media and social media mining, I further propose an innovative method to customize news event profiling with massive Twitter data. A propagation model simultaneously exploring global and local context was developed on a tripartite graph where news sentences and tweets are bridged by content units. Content units enables fine-grained quality information transfer between news and tweets so that the social impact in Twitter is propagated to news. Extensive data analysis and a comprehensive user study demonstrated the effectiveness of the proposed method.

References

[1] Deepak Agarwal and Bee-Chung Chen. Regression-based latent factor models. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, New York, NY, USA, 2009. ACM.

[2] Albert Angel, Nikos Sarkas, Nick Koudas, and Divesh Srivastava. Dense subgraph maintenance under streaming edge weight updates for real-time story identification. *Proceedings of the VLDB Endowment*, 5(6):574–585, February 2012. ISSN 2150-8097. doi: 10.14778/2168651.2168658. URL http://dx.doi.org/10.14778/2168651.2168658.

[3] Luciano Barbosa and Junlan Feng. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10 Posters, pages 36–44, 2010.

[4] Yoshua Bengio. Learning deep architectures for ai. *Foundations and Trends in Machine Learning*, 2(1):1–127, January 2009. ISSN 1935-8237. doi: 10.1561/2200000006. URL http://dx.doi.org/10.1561/2200000006.

[5] Adam L Berger, Vincent J Della Pietra, and Stephen A Della Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1): 39–71, 1996.

[6] David M. Blei and John D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 113–120, New York, NY, USA, 2006. ACM. ISBN 1-59593-383-2. doi: 10.1145/1143844.1143859. URL http://doi.acm.org/10.1145/1143844.1143859.

[7] David M. Blei, Thomas L. Griffiths, and Michael I. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM)*, 57(2):7:1–7:30, February 2010. ISSN 0004-5411. doi: 10.1145/1667053.1667056.

[8] Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 335–336, 1998.

[9] Chen Cheng, Haiqin Yang, Irwin King, and Michael R. Lyu. Fused matrix factorization with geographical and social influence in location-based social networks. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, AAAI '12, 2012.

[10] Hong Cheng, Jihang Ye, and Zhe Zhu. What's your next move: User activity prediction in location-based social networks. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, SDM '13, pages 171–179, 2013.

[11] Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility: user movement in location-based social networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 1082–1090, 2011.

[12] Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*, 2013.

[13] Hongbo Deng, Michael R Lyu, and Irwin King. A generalized co-hits algorithm and its application to bipartite graphs. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 239–248. ACM, 2009.

[14] Li Deng and Dong Yu. Deep learning: Methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387, 2014.

[15] Günes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, pages 457–479, 2004.

[16] Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1606–1611, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc. URL http://dl.acm.org/citation.cfm?id=1625275.1625535.

[17] Huiji Gao and Huan Liu. Location-based social network data repository, 2014. URL http://www.public.asu.edu/~hgao16/dataset.html.

[18] Huiji Gao, Jiliang Tang, and Huan Liu. Exploring social-historical ties on location-based social networks. In *Proceedings of the Sixth International Conference on Weblogs and Social Media*, ICWSM '12, 2012.

[19] Huiji Gao, Jiliang Tang, and Huan Liu. Mobile location prediction in spatio-temporal context. In *Nokia Mobile Data Challenge 2012 Workshop*, MDC '12, 2012.

[20] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. Modeling temporal effects of human mobile behavior on location-based social networks. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, CIKM '13, pages 1673–1678, 2013.

[21] Huiji Gao, Jiliang Tang, Xia Hu, and Huan Liu. Exploring temporal effects for location recommendation on location-based social networks. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pages 93–100, 2013.

[22] Wei Gao, Peng Li, and Kareem Darwish. Joint topic modeling for event summarization across news and social media streams. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 1173–1182, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1156-4. doi: 10.1145/2396761.2398417. URL http://doi.acm.org/10.1145/2396761.2398417.

[23] Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. Technical report, Stanford University, 2009.

[24] Weiwei Guo, Hao Li, Heng Ji, and Mona T. Diab. Linking tweets to news: A framework to enrich short text data in social media. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, ACL '13, pages 239–249, 2013.

[25] Tuan-Anh Hoang-Vu, Aline Bessa, Luciano Barbosa, and Juliana Freire. Bridging vocabularies to link tweets and news. *International Workshop on the Web and Databases, WebDB*, 2014.

[26] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106 (4):620, 1957.

[27] Hyungsul Kim, Yizhou Sun, Julia Hockenmaier, and Jiawei Han. Etm: Entity topic models for mining documents associated with entities. In *Proceedings of the 2012 IEEE International Conference on Data Mining*, ICDM '12, pages 349–358, Washington, DC, USA, 2012. IEEE Computer Society. ISBN 978-0-7695-4905-7. doi: 10.1109/ICDM.2012.107. URL http://dx.doi.org/10.1109/ICDM.2012.107.

[28] Alok Kothari, Walid Magdy, Kareem Darwish, Ahmed Mourad, and Ahmed Taei. Detecting comments on news articles in microblogs. In *Proceedings of the Seventh International Conference on Weblogs and Social Media*, ICWSM '13, 2013.

[29] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, pages 79–86, 1951.

[30] Takeshi Kurashima, Tomoharu Iwata, Takahide Hoshide, Noriko Takaya, and Ko Fujimura. Geo topic model: Joint modeling of user's activity area and interests for location recommendation. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, pages 375–384, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1869-3. doi: 10.1145/2433396.2433444. URL http://doi.acm.org/10.1145/2433396.2433444.

[31] J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[32] Min Li, Jingjing Wang, Wenzhu Tong, Hongkun Yu, Xiuli Ma, Yucheng Chen, Haoyan Cai, and Jiawei Han. EKNOT: event knowledge from news and opinions in twitter. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pages 4367–4368, 2016. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11839.

[33] Bin Liu and Hui Xiong. Point-of-interest recommendation in location based social networks with topic and location awareness. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, SDM '13, pages 396–404, 2013.

[34] Bin Liu, Yanjie Fu, Zijun Yao, and Hui Xiong. Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 1043–1051, 2013.

[35] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528, 1989.

[36] Carmen De Maio, Giuseppe Fenza, Vincenzo Loia, and Mimmo Parente. Online query-focused twitter summarizer through fuzzy lattice. In *2015 IEEE International Conference on Fuzzy Systems*, FUZZ-IEEE '15, pages 1–8, 2015. doi: 10.1109/FUZZ-IEEE.2015.7337927.

[37] Tomonari Masada, Senya Kiyasu, and Sueharu Miyahara. Comparing lda with plsi as a dimensionality reduction method in document clustering. In *Large-Scale Knowledge Resources. Construction and Application*, pages 13–26. Springer, 2008.

[38] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of 2013 Conference on Advances in Neural Information Processing Systems*, NIPS '13, pages 3111–3119, 2013.

[39] David Newman, Chaitanya Chemudugunta, and Padhraic Smyth. Statistical entity-topic models. In *Proceedings of the 12nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 680–686, New York, NY, USA, 2006. ACM. ISBN 1-59593-339-5. doi: 10.1145/1150402.1150487.

[40] David Newman, Edwin V. Bonilla, and Wray L. Buntine. Improving topic coherence with regularized topic models. In *Proceedings of 2011 Conference on Advances in Neural Information Processing Systems*, NIPS '11, pages 496–504, 2011.

[41] Ariadna Quattoni, Sybor Wang, Louis-Philippe Morency, Michael Collins, and Trevor Darrell. Hidden conditional random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(10):1848, 2007.

[42] Dragomir R Radev, Hongyan Jing, and Malgorzata Budzikowska. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic Summarization*, pages 21–30. Association for Computational Linguistics, 2000.

[43] Jonathon Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, ACLstudent '05, pages 43–48, 2005.

[44] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 851–860, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-799-8. doi: 10.1145/1772690.1772777. URL http://doi.acm.org/10.1145/1772690.1772777.

[45] Jagan Sankaranarayanan, Hanan Samet, Benjamin E. Teitler, Michael D. Lieberman, and Jon Sperling. Twitterstand: News in tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, pages 42–51, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-649-6. doi: 10.1145/1653771.1653781.

[46] Lidan Shou, Zhenhua Wang, Ke Chen, and Gang Chen. Sumblr: Continuous summarization of evolving tweet streams. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 533–542, New York, NY, USA, 2013. ACM.

[47] Yangqiu Song and Dan Roth. On dataless hierarchical text classification. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, AAAI '14, 7 2014.

[48] Ilija Subavsic and Bettina Berendt. Peddling or creating? investigating the role of twitter in news reporting. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR'11, pages 207–213, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-20160-8.

[49] Jian Tang, Ming Zhang, and Qiaozhu Mei. One theme in all views: Modeling consensus topics in multiple contexts. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 5–13, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2174-7. doi: 10.1145/2487575.2487682.

[50] Yee Whye Teh. A hierarchical bayesian language model based on pitman-yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL '06, pages 985–992. Association for Computational Linguistics, 2006.

[51] Manos Tsagkias, Maarten de Rijke, and Wouter Weerkamp. Linking online news and social media. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM '11, pages 565–574, New York, NY, USA, 2011. ACM.

[52] Jan Vosecky, Di Jiang, Kenneth Wai-Ting Leung, and Wilfred Ng. Dynamic multifaceted topic discovery in twitter. In *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '13, pages 879–884, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2263-8. doi: 10.1145/2505515.2505593. URL http://doi.acm.org/10.1145/2505515.2505593.

[53] Tadej Štajner, Bart Thomee, Ana-Maria Popescu, Marco Pennacchiotti, and Alejandro Jaimes. Automatic selection of social media responses to news. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 50–58, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2174-7. doi: 10.1145/2487575.2487659. URL http://doi.acm.org/10.1145/2487575.2487659.

[54] Chi Wang, Marina Danilevsky, Jialu Liu, Nihit Desai, Heng Ji, and Jiawei Han. Constructing topical hierarchies in heterogeneous information networks. In *Proceedings of the 2013 IEEE International Conference on Data Mining*, ICDM '13, pages 767–776, 2013. doi: 10.1109/ICDM.2013.53. URL http://dx.doi.org/10.1109/ICDM.2013.53.

[55] Jingjing Wang, Wenzhu Tong, Hongkun Yu, Min Li, Xiuli Ma, Haoyan Cai, Tim Hanratty, and Jiawei Han. Mining multi-aspect reflection of news events in twitter: Discovery, linking and presentation. In *Proceedings of the 2015 IEEE International Conference on Data Mining*, ICDM '15, pages 429–438. IEEE, 2015.

[56] Jingjing Wang, Min Li, Jiawei Han, and Xiaolong Wang. Modeling check-in preferences with multidimensional knowledge: A minimax entropy approach. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM '16, pages 297–306, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3716-8. doi: 10.1145/2835776.2835839. URL http://doi.acm.org/10.1145/2835776.2835839.

[57] Shaojun Wang, Dale Schuurmans, and Yunxin Zhao. The latent maximum entropy principle. *ACM Transactions on Knowledge Discovery from Data*, 6(2):8:1–8:42, July 2012. ISSN 1556-4681. doi: 10.1145/2297456.2297460. URL http://doi.acm.org/10.1145/2297456.2297460.

[58] Xuerui Wang and Andrew McCallum. Topics over time: A non-markov continuous-time model of topical trends. In *Proceedings of the 12nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 424–433, New York, NY, USA, 2006. ACM. ISBN 1-59593-339-5. doi: 10.1145/1150402.1150450.

[59] Zhongyu Wei and Wei Gao. Utilizing microblogs for automatic news highlights extraction. In *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers*, COLING '14, pages 872–883, 2014.

[60] Zhongyu Wei and Wei Gao. Gibberish, assistant, or master?: Using tweets linking to news for extractive single-document summarization. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 1003–1006. ACM, 2015.

[61] Zhongyu Wei, Yang Liu, Chen Li, and Wei Gao. Using tweets to help sentence compression for news highlights generation. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, ACL '15, 2015.

[62] Hongzhi Yin, Yizhou Sun, Bin Cui, Zhiting Hu, and Ling Chen. Lcars: a location-content-aware recommender system. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 221–229. ACM, 2013.

[63] Chun-Nam John Yu and Thorsten Joachims. Learning structural svms with latent variables. In *Proceedings of the 26th International Conference on Machine Learning*, ICML '09. ACM, 2009.

[64] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat-Thalmann. Time-aware point-of-interest recommendation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 363–372, 2013.

[65] Evi Yulianti, Sharin Huspi, and Mark Sanderson. Tweet-biased summarization. *Journal of the Association for Information Science and Technology*, 2015.

[66] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. Comparing twitter and traditional media using topic models. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, ECIR'11, pages 338–349, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-20160-8. URL http://dl.acm.org/citation.cfm?id=1996889.1996934.

[67] Vincent Wenchen Zheng, Yu Zheng, Xing Xie, and Qiang Yang. Collaborative location and activity recommendations with gps history data. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 1029–1038, 2010.

[68] Yu Zheng and Xing Xie. Learning location correlation from gps trajectories. In *Mobile Data Management*, pages 27–32, 2010.

[69] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining correlation between locations using human location history. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, pages 472–475, 2009.

[70] Yu Zheng, Lizhu Zhang, Zhengxin Ma, Xing Xie, and Wei-Ying Ma. Recommending friends and locations based on individual location history. *ACM Transactions on the Web (TWEB)*, 5(1):5, 2011.

[71] Dengyong Zhou, John C. Platt, Sumit Basu, and Yi Mao. Learning from the wisdom of crowds by minimax entropy. In *Proceedings of 2012 Conference on Advances in Neural Information Processing Systems*, NIPS '12, pages 2204–2212, 2012.

[72] Jun Zhu, Eric P. Xing, and Bo Zhang. Partially observed maximum entropy discrimination markov networks. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Proceedings of 2009 Conference on Advances in Neural Information Processing Systems*, NIPS '09. Curran Associates, Inc., 2009.

[73] Song Chun Zhu, Ying Nian Wu, and David Mumford. Minimax entropy principle and its application to texture modeling. *Neural Computation*, 9(8):1627–1660, 1997.