Linking fine-grained locations in user comments

Han, Jialong; Sun, Aixin; Cong, Gao; Zhao, Wayne Xin; Ji, Zongcheng; Phan, Minh C.

2017

Han, J., Sun, A., Cong, G., Zhao, W. X., Ji, Z., & Phan, M. C. (2018). Linking fine-grained locations in user comments. *IEEE Transactions on Knowledge and Data Engineering*, 30(1), 59-72. doi:10.1109/TKDE.2017.2758780

<https://hdl.handle.net/10356/86191>

<https://doi.org/10.1109/TKDE.2017.2758780>

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: <https://doi.org/10.1109/TKDE.2017.2758780>.

Downloaded on 26 Apr 2023 17:31:06 SGT

Linking Fine-Grained Locations in User Comments

Jialong Han, Aixin Sun, Gao Cong, Wayne Xin Zhao, Zongcheng Ji, and Minh C. Phan

Abstract—Many domain-specific websites host a profile page for each entity (e.g., locations on Foursquare, movies on IMDb, and products on Amazon) for users to post comments on. When commenting on an entity, users often mention other entities for reference or comparison. Compared with web pages and tweets, the problem of disambiguating the mentioned entities in user comments has not received much attention. This paper investigates linking fine-grained locations in Foursquare comments. We demonstrate that the *focal location*, i.e., the location that a comment is posted on, provides rich contexts for the linking task. To exploit such information, we represent the Foursquare data in a graph, which includes locations, comments, and their relations. A probabilistic model named *FocalLink* is proposed to estimate the probability that a user mentions a location when commenting on a focal location, by following different kinds of relations. Experimental results show that *FocalLink* is consistently superior under different collective linking settings.

Index Terms—Entity Linking, Named Entity Recognition, Point-of-Interest, User Comment, Knowledge Base.

1 INTRODUCTION

WITH the prevalence of GPS-enabled Internet access devices such as smartphones and tablets, people are more willing to share and query information about locations. Thanks to the contributions from millions of users, location-based social networks (LBSNs) like Foursquare¹, Yelp², and Google Maps, have accumulated huge amount of information about *fine-grained locations* in the form of comments/tips/reviews, ratings, geographical annotations, photos, and others. Here, a fine-grained location may be a restaurant, a shopping mall, a park, a landmark building, or other kinds of point-of-interests.

Typically, on LBSNs, each location has a dedicated *profile page*. A user can open the profile page of a location to view information about it, or post comments on it. In Figure 1, we exemplify this with data from Foursquare. In this figure, locations are represented by ellipse nodes, e.g., Jurong Point and IMM Building, which are two shopping malls in Singapore. Location profile pages are indicated by rectangular boxes, where comments on the respective locations are posted. For instance, on the page of IMM Building, a user left a comment saying “Go for *daiso*, 3rd fl, the 2dollar shop”. For clarity, given a comment, we refer to the location being commented on (e.g., IMM Building in this example) as the *focal location*.

When commenting on the focal location, users may also mention other locations, which are marked in grey in Figure 1. To distinguish them from focal locations, we refer to the locations mentioned in comments as *mentioned locations*. Mentioned locations may be the focal locations themselves,

Fig. 1. A motivating example of Foursquare data graph. For the clarity of this figure, not all nodes and edges are presented.

or other related ones. For instance, in the first comment on Jurong Point, the focal location itself is mentioned by an anchor (or surface form) *Jurong pt*. Meanwhile, in the aforementioned comment on IMM Building, a chain miscellaneous store named *daiso* located inside IMM Building is mentioned. According to manual annotation results from a sample of 4,000 Foursquare comments, only 18.1% (150 out of 828) of mentioned locations are the focal locations themselves. In other words, most of the time, users mention related locations other than focal locations themselves.

For various types of texts like web pages and tweets, *entity linking* [1] has proved useful in facilitating understanding and searching those texts, as well as extracting information from them. In those texts, given a detected anchor that refers to some entity, an entity linker resolves the ambiguity of the anchor by mapping it to the right entry in some database (e.g., Wikipedia). In our case, if mentioned locations could be linked to the right location pages (like in Figure 1), the following applications could be achieved or enhanced:

- **Comment gathering.** Business operators and shop

- J. Han, A. Sun, G. Cong, Z. Ji, and M. Phan are with School of Computer Science and Engineering, Nanyang Technological University, Singapore. E-mails: jialonghan@gmail.com, axsun@ntu.edu.sg, gaocong@ntu.edu.sg, jizongcheng@gmail.com, phan0050@e.ntu.edu.sg
- W. X. Zhao is with School of Information, Renmin University of China. E-mail: batmanfly@gmail.com

1. <http://www.foursquare.com/>
2. <http://www.yelp.com/>

owners have fully realized the importance of user comments on social platforms. However, monitoring comments on their own location profile pages is far from enough because the location could be mentioned elsewhere as in the *daiso* example. A simple text search with the name is possible, but it may return inexact results due to ambiguity. Location linking can help to effectively gather all comments about a specific location no matter where they were posted.

- **Sentiment analysis.** After linking mentioned locations, sentiment analysis can be conducted in a more precise manner, because not all comments on a profile page are meant for the focal location. Users may express negative sentiments on a mentioned location which is not the focal location.
- **Location recommendation.** From users' perspective, mentioning another location rather than the focal location often means that the two locations are related in some sense, *e.g.*, of similar types, offering similar services, or located close to each other. This could be a strong signal to exploit in (next) location recommendation.

Besides the above benefits, we note that our study may be generalizable to other domains like movies (*e.g.*, IMDb) and e-commerce (*e.g.*, Amazon). Websites in those domains also possess similar structures as in Figure 1, where each entity has a profile page to receive comments. We argue that investigations in location domain may give rise to techniques generalizable to the same linking problem in other domains.

Although entity linking for formal documents like web pages has been relatively well studied, efforts on the same task for user comments remain limited. Entity linking for user comments not only faces the ambiguity of mentions. Like tweets, it is also rendered more challenging by the short nature of comments.³ For example, given that *Daiso* has more than ten branches in Singapore, it is difficult to judge solely from the short comment "Go for *daiso* ..." in the beginning of this section which one it is referring to, for the text contains little contextual information.

Because text information is scarce in user comments, any extra information should be exploited to assist with the task. Luckily, for a given comment, the focal location is always available and unambiguous, which provides important contextual information. For example, if we know that the aforementioned comment about *Daiso* is posted on the page of *IMM Building*, the "*daiso*" here tends to refer to the branch inside *IMM Building*. This is because their spatial containment *relation* is indicated by "3rd fl" (the third floor) in the comment.

In this paper, we present *FocalLink*, an unsupervised linking model, which exploits focal locations as additional clues to link locations in user comments. We view all locations and their rich relations, *e.g.*, space containments, type overlaps, and geographical distance, as a graph like Figure 1. When a user comments on a focal location, say, *IMM Building*, we view the user as having finished a

random walk on the graph starting with *IMM Building*. This walk may follow certain relation and ends with *Daiso*, the mentioned location. By adopting this assumption, we expect to encode the contextual role of focal locations. We also observe that the unknown relation followed by a user is implied by some indicative words in the comment. Through modeling the generative process of comments, we seek to learn good alignments between those indicative words and relations on the graph, *e.g.*, "3rd fl" and spatial containment. By achieving this, candidate locations linked to the focal location via relations supported by the comment could be given more credit.

Finally, *collective linking* [1] has been reported effective by simultaneously linking multiple mentions. Observing that a profile page may contain multiple comments and that a comment may mention multiple locations, we also investigate two different settings to apply collective linking in our problem.

We summarize the contributions of this paper as follows:

- We address linking fine-grained locations in user comments, which may facilitate an effective use of user generated content and inspire similar tasks in other domains.
- We propose a probabilistic linking model to exploit extra contextual information brought by focal locations.
- We experimentally validate the superiority of our model under different collective linking settings.

2 RELATED WORK

Named Entity Recognition (NER) has been investigated for several decades [2]. For formal documents, satisfactory performance could be achieved by state-of-the-art machine learning algorithms (*e.g.*, CRF) with external dictionaries and comprehensive linguistic features, *e.g.*, Part-of-Speech (POS) tags and capitalizations [3]. Although newer than NER, the Entity Linking (EL) problem on formal documents has also been studied for a decade [4]. Given a detected anchor, various cues may be considered to link the anchor, *e.g.*, the context words and other anchors in the same document. For more literatures on both tasks, we refer readers to the two extensive surveys [5], [6].

Named entity recognition for tweets. Although NER is not the focus of this paper, it is inevitably involved as preparations for linking. Therefore, we review some previous NER efforts on tweets, which, like comments, are a significant type of informal and short user generated content. When facing tweets, traditional NER methods are at risk of deteriorated performance. Because of the informal writing, NER for tweets may suffer from Out-Of-Vocabulary (OOV) words as well as unreliable linguistic features. Ritter *et al.* [7] retrained the entire NLP pipeline for NER of tweets. Brown clustering was used to improve POS tags of OOV words, and a dedicated classifier was trained to recognize whether the capitalization in a tweet is informative. Compared with StanfordNER, a state-of-the-art NER tool for general documents, the twitter-specific NER pipeline achieved a 52% increase in F_1 . Liu *et al.* [8] trained a tweet normalization

3. 15 words on average, according to our Foursquare dataset consisting of 0.44 million comments on Singaporean locations.

model to correct OOV words (*e.g.*, “goood”) before performing NER. To facilitate the processing of hard cases, a k-nearest-neighbor (KNN) classifier was also designed to provide the CRF classifier with global information, *e.g.*, how a word is labeled in other tweets. Li *et al.* [9] addressed a novel streaming setting of tweet NER. They exploited the gregarious property of entity segments in a twitter stream to rank entity segments.

Entity linking for tweets. When applied to tweets, conventional EL methods are also challenged by insufficient information because of the short length of tweets. To resolve the lack of information, efforts on tweet entity linking mainly concentrate on exploiting additional contexts, whether explicit or implicit. On one hand, explicit contexts are directly retrievable, *e.g.*, posting time and locations of tweets, and social connections of tweet users. On the other hand, implicit contexts need to be modeled and estimated, such as tweet users’ interests or global entity linking history within a recent time window. Fang *et al.* [10] exploited spatio-temporal information of tweets to assist entity linking. Specifically, an entity prior w.r.t. time and location is estimated and used to replace the coarse-grained global popularity information. Hua *et al.* [11] considered a user’s social connections and entity recency in tweet EL. They assumed that an entity is more likely to be referred to if it belongs to a recent hot topic, or the tweet author’s followee also mentions it. Shen *et al.* [12] modeled user interest as an additional context for linking. For example, if a user often tweets on sports, an anchor “Michael Jordan” in his tweet is likely to refer to the basketball star rather than the machine learning researcher. Davis *et al.* [13] studied linking entities in tweets in a streaming setting. Liu *et al.* [14] explored entity linking for tweets. Observing that some entities are often mentioned in many tweets, they compute a mention-mention similarity to disambiguate hard ones using information from easy ones. Some recent studies also considered jointly performing entity recognition and linking, to enable information to propagate in both directions [15], [16], [17], [18], [19].

Entity recognition and linking for other short texts. User comments or reviews are widely available across Web 2.0 sites in various domains like locations, movies, and products. Although opinion mining and sentiment analysis [20], [21] have been widely studied on such short comments, entity recognition and linking were not given enough attention. Ren *et al.* [22] studies entity recognition and typing on Yelp reviews. Xu *et al.* [23] investigates extracting names of complementary products that may potentially work together with focal products. However, neither of them considered linking. Recently, entity recognition and linking were also studied for keyword queries [24], [25], [26] and proved to improve query understanding and document retrieval [27]. To the best of our knowledge, we are the first to address entity linking in user comments. The common characteristics between tweets and user comments inspire us to investigate whether EL on comments is trivial, given the extensive studies of tweet EL. Similar to tweet EL, our approach falls into the line of studies which exploits new contextual linking cues.

Collective entity linking. For studies on tweets and general documents [10], [11], [12], [14], [17], [18], [28], [29], [30], entity anchors are linked to cross-domain knowledge bases like Wikipedia and YAGO, which contains rich information to assist linking. The rich linkages between concepts in both knowledge bases enabled various methods based on collective linking [1], [18], [30], [31]. In the academic domain, Shen *et al.* [32] used the DBLP network to resolve ambiguous author names. By leveraging meta-paths [33], they resorted to unambiguous entities in the same document like co-authors and publishing venues to help linking. In our study, collective linking may not apply to comments with only one mentioned location, which is common due to the short length of comments. We resort to modeling focal locations as additional contexts, which is always available for user comments. In experiments, we will demonstrate the above by instantiating two collective linking variants for all compared approaches.

Location recognition and linking. As described above, entity recognition and linking are usually studied in open domains. Public challenges like MUC (Message Understanding Conference [34], [35]) and ACE (Automatic Content Extraction [36]) have been involving entity recognition and typing for two decades, including locations. In addition, annotation languages [37] and ontology [38] are proposed for spatial language. Besides location entities, they also capture spatial relations expressed in natural language, *e.g.*, “3rd fl of”. Such language phenomenon is explained in the *figure-ground theory* [39]. Different from challenges (*e.g.*, [36]) or studies (*e.g.*, [40]) on recognizing such relations, we only model spatial relations on the data graph, which are also expressed in user comments, to assist linking. Our ultimate goal is to link entities rather than recognize relations. As for linking, there have been efforts [41], [42] on building benchmark platforms for EL on Wikipedia. They implement Wikipedia-specific techniques like anchor text prior and semantic relatedness, which are not applicable to our Foursquare dataset. Therefore, we do not use them in our study. In the location domain, location linking is also studied in the name of *toponym resolution* [43]. Lieberman *et al.* [44] and Adelfio *et al.* [45] considered proximity, sibling, and category features to link multiple locations in a common context. Their solutions, together with Shen’s [32], could be regarded as other forms of collective linking. In [46], Dalvi *et al.* studied the problem of matching tweets to restaurants. The distance between a restaurant and the location where a tweet is sent out is their major consideration. However in our problem, besides the distance between locations, their spatial containment and type information are also useful and worth modeling.

3 PRELIMINARIES

3.1 Problem Definition

In this section, we begin with concepts and their notations, after which we formulate the problem of location linking in user comments. Table 1 summarizes primary notations used throughout this paper.

Location Database and Data Graph. To facilitate location linking, we need a database $\mathcal{E}$ consisting of all locations

TABLE 1
Frequently used notations.

Notation	Definition and description
$e \in \mathcal{E}$	An arbitrary location e from a database $\mathcal{E}$
$\mathcal{C} = \langle e_f, c_0 \rangle$	A user comment $\mathcal{C}$ with focal location e_f and comment text c_0
$m = \langle e_f, a, c \rangle$	A mention m with focal location e_f , anchor a , and surrounding context c
$w \in c$	An arbitrary word w in surrounding context c
$\mathcal{E}(m)$	Candidates generated for m
d_e and D	Virtual document for e and the global virtual document
$\mathcal{G}$	The data graph as in Figure 1
$r \in R$	An arbitrary relation r on graph $\mathcal{G}$
π and π_r	A prior distribution over R and the prior probability of r
θ_r	Relation word distribution of relation r
s and S	Switch variable(s) for the mixture model that generates w or c from d_e , θ_r , and D
λ , λ_1 , and λ_2	Mixture weights λ where λ_1 , λ_2 , and $(1 - \lambda_1 - \lambda_2)$ are for d_e , θ_r , and D , respectively

we are concerned with. The database here is similar to the collection of entities in Wikipedia as in the “linking to Wikipedia” problem. More specifically, the database contains fine-grained locations collected from Foursquare within a predefined geographical area. We use $e \in \mathcal{E}$ to denote an arbitrary location in the database, and $e.name$ to denote its full name. Besides location names, statistics like numbers of historical check-ins and location descriptions are also stored in $\mathcal{E}$.

By taking spatial containments, location types, and distance information into consideration, we also view the data as a graph $\mathcal{G}$ (see Figure 1). Note that, graph is a flexible data structure, and the data graph shown in Figure 1 can be easily extended to include other types of nodes and relations, *e.g.*, user nodes and the co-checkin relationship between two locations, if such information is available. To be detailed in Section 4, the data graph $\mathcal{G}$ helps us to address location linking by modeling the possible relations users may follow to mention a location while commenting on another.

User Comment. A user comment $\mathcal{C}$ is a tuple $\langle e_f, c_0 \rangle$. Here e_f is the focal location, and c_0 is the text of the comment. Note that we assume that the focal location e_f is known for a user comment. That is, given the comment “Go for daiso, 3rd fl, the 2dollar shop” from Figure 1, we know that this comment was posted on the profile page of IMM Building (*i.e.*, the focal location e_f).

Mention. Suppose a comment $\mathcal{C} = \langle e_f, c_0 \rangle$ mentions a location in its text c_0 . We call the text segment in c_0 that corresponds to the mentioned location an *anchor*, and denote it by a . The other words in the comment text are *surrounding context*, denoted by c . Finally, we define $m = \langle e_f, a, c \rangle$ as a *mention*. For example, in comment $\mathcal{C} = \langle \text{IMM Building}, \text{“Go for daiso, 3rd fl, the 2dollar shop”} \rangle$, a mention m is $\langle \text{IMM Building}, \text{“daiso”, } \{\text{Go, for, 3rd, fl, the, 2, dollar, shop}\} \rangle$. In this work, we assume that all mentions are detected beforehand and given as input to the location linking problem. This could be done by leveraging state-of-the-art NER tools, *e.g.*, Conditional Random Field (CRF) [47].

Fig. 2. An overview of our solution. Dotted boxes and arrows are new linking cues to be investigated in this paper.

PROBLEM DEFINITION. Given a user comment $\mathcal{C} = \langle e_f, c_0 \rangle$ and all location mentions $\{m\}$ detected from $\mathcal{C}$, we perform the following task. For each mention m , we find the location entry $e^{(m)} \in \mathcal{E}$ being referred to. In the case that m ’s ambiguity cannot be resolved or $e^{(m)}$ is not listed in $\mathcal{E}$, m should be linked to NULL, which denotes “unlinkable”. For example, the above mention should be linked to the Daiso branch in IMM Building.

3.2 Solution Overview and Baselines

In this section, we give an overview of our solution. We decompose the location linking task into two sub-tasks, namely *candidate generation* and *candidate ranking*, as illustrated in Figure 2. In the first stage, locations potentially matching the mention m are retrieved. They are fed to the second stage, which ranks them and links m to the top ranked location. In the following, we debrief our strategies for candidate generation, and describe three baseline methods for candidate ranking.

3.2.1 Candidate Location Generation

As input, mentions m in user comment $\mathcal{C}$ are given. However, the location database $\mathcal{E}$ may be potentially large, making it unwise to compare m with every entry $e \in \mathcal{E}$ in order to link m . Therefore, we employ a candidate generation component. It retrieves for each m a much smaller set of locations $\mathcal{E}(m)$ by considering only literal match between a and full names $e.name$ of locations $e \in \mathcal{E}$. We adopt the following matching rules between a and $e.name$. A location e meeting any of the rules will be added to $\mathcal{E}(m)$.

- 1) The weighted Jaccard similarity between a and $e.name$ is larger than a threshold τ , *e.g.*, *Charles n Keith* and *Charles & Keith*;
- 2) When all spaces are ignored, a is a prefix of $e.name$, *e.g.*, *Pu Tien* and *PUTIEN Restaurant*;
- 3) The words of a are fully covered by those of $e.name$ *e.g.*, *Transformers ride* and *Transformers The Ride: The Ultimate 3D Battle*; and
- 4) a is an all-capitalized word, and is an acronym of $e.name$, *e.g.*, *GWC* and *Great World City*.

In Section 6.1.3, we will show that the above four rules capture the ground truth locations of 83.7% mentions. We note that this component could be further improved by more sophisticated techniques. However, it is not the key focus of this paper.

TABLE 2
Examples of relations between the focal and mentioned locations. Anchors of mentioned locations are underlined.

Name	Illustration on $\mathcal{G}$	Description	An example pair of (focal locations, "comments")
Self-Ref	$e_f = e$	e_f is identical to e .	$\langle \text{Zouk (a nightclub)}, \text{"A visit to Zouk is a must for anyone in Singapore..."}$
Inside	$e_f \xrightarrow{\text{inside}} e$	e_f is located inside e .	$\langle \text{The Manhattan Fish Market (a restaurant)}, \text{"6th floor of Plaza Singapura..."}$
Contains	$e_f \xleftarrow{\text{inside}} e$	e_f spatially contains e .	$\langle \text{Gourmet Paradise (a food court)}, \text{"Must try Soon Heng Rojak."}$
Co-Inside	$e_f \xrightarrow{\text{inside}} o \xleftarrow{\text{inside}} e$	e_f and e are in the same building.	$\langle \text{The Cathay Restaurant}, \text{"...next to it is the very interesting Cathay Gallery..."}$
Co-Type	$e_f \xrightarrow{\text{type}} o \xleftarrow{\text{type}} e$	e_f and e share the same type.	$\langle \text{Royal Plaza On Scotts (a hotel)}, \text{"Better than MBS."}$
Near	$e_f \xleftrightarrow{2km} e$	e_f and e are within 3km.	$\langle \text{The Fullerton Hotel}, \text{"... great view across to Marina Bay Sands."}$

3.2.2 Candidate Ranking Baselines

After candidates $\mathcal{E}(m)$ are retrieved, the linking of m is to determine the most probable $e \in \mathcal{E}(m)$ or NULL. We estimate the conditional probability $P(e|m)$, and link m to

$$e^{(m)} = \arg \max_{e \in \mathcal{E}(m)} P(e|m)$$

As for NULL links, we adopt the following simple strategy: m is linked to NULL only when $\mathcal{E}(m)$ is empty. There may be other approaches to handle NULL links like dummy entry [28] and thresholding [30]. We leave those alternatives for future study. Next, we concentrate on three baseline methods that instantiate $P(e|m)$ with different kinds of information available in m .

Popularity. In this method, we rank candidate locations for a mention based on their popularity. By assuming that locations with higher popularity are more likely to be mentioned, we link a mention m to $e^{(m)}$ by

$$e^{(m)} = \arg \max_{e \in \mathcal{E}(m)} P(e) \quad (1)$$

Here the popularity $P(e)$ of e is estimated by the number of user check-ins at this location.

PopContext. The first baseline exploits only the popularity information. In addition to the popularity information, PopContext utilizes the set of context words c as well. We link a mention m to $e^{(m)}$ by

$$\begin{aligned} e^{(m)} &= \arg \max_{e \in \mathcal{E}(m)} P(e|c) \\ &= \arg \max_{e \in \mathcal{E}(m)} P(c|e)P(e). \end{aligned} \quad (2)$$

Here $P(c|e)$ is the conditional probability that a user wraps the anchor of e with context words c . Following Han *et al.* [28], we estimate $P(c|e)$ by the unigram probabilistic language model [48] as follows:

$$P(c|e) = \prod_{w \in c} [\lambda_1 P(w|d_e) + (1 - \lambda_1)P(w|D)]. \quad (3)$$

Here each location e is viewed as a virtual document d_e consisting of its description text and all comments on its profile page. A global virtual document $D = \cup_{e \in \mathcal{E}} d_e$ is built by concatenating all virtual documents. Given a context word w , $P(w|d_e)$ and $P(w|D)$ are estimated by the relative frequency of w in d_e and D , respectively. λ_1 is an interpolation parameter for smoothing $P(w|d_e)$ with $P(w|D)$.

PopContextDist. The above two baselines does not exploit the focal entity e_f . A user often comments on e_f simply

because she is at this place, thus tending to mention nearby locations. We embed this observation by taking e_f into consideration and substituting $P(e)$ in Eq. 2 with a *location-sensitive popularity* $P(e|e_f)$:

$$\begin{aligned} e^{(m)} &= \arg \max_{e \in \mathcal{E}(m)} P(e|c, e_f) \\ &= \arg \max_{e \in \mathcal{E}(m)} P(c|e)P(e|e_f). \end{aligned} \quad (4)$$

Here $P(e|e_f)$ is defined as

$$P(e|e_f) = \frac{1}{Z_{e_f}} nDist(e, e_f)P(e) \quad (5)$$

$$\text{where } nDist(e, e') = \max\{0, 1 - \frac{Dist(e, e')}{Maxdist}\}. \quad (6)$$

In Eq. 6, $nDist(e, e')$ is the normalized geographical distance between two locations e and e' with normalizer $Maxdist$. In Eq. 5, Z_{e_f} is a normalizing factor to make the probability sum to one. Intuitively, Eq. 5 assumes a circular area centered at e_f with radius $Maxdist$. In this area, the more popular e is, the nearer it is to e_f , the more probably it will be mentioned.

4 FOCALLINK: EXPLOITING FOCAL LOCATIONS

In the above baselines, different kinds of information available in a mention are involved. However, when commenting on e_f , users do not mention a location only because of distance and popularity, but also for other reasons. Moreover, words in the surrounding context are not always about the mentioned entity itself. In this section, we propose FocalLink, an unsupervised approach to model relations between the focal and mentioned locations, as well as the context words caused by the relations.

4.1 Observations and Assumptions

Given a mention $m = \langle e_f, a, c \rangle$ and an entity e to be mentioned by a user, we present **two observations** on the interactions within m , and between m and e . Based on these two observations, we design our model FocalLink.

Observation 1: When commenting on location e_f , a user may mention another related location e . Possible underlying relations r between e_f and e are, e.g., being of similar types, near to each other, or one is located inside another, to name a few. Those relations depend on basic information from Foursquare, e.g., the types of, the coordinates of, and spatial containments between, locations. To represent possible relations, we model the above basic information as edges in the

Step 3: Write the surrounding context. In this step, the user wraps anchor a with some words about e , e_f , and r as the surrounding context c . By **Observation 2**, the probability $P(c|e_f, e, r)$ of writing down c depends not only on e_f and e , but also on r . In baseline PopContext, we assume that the probability of writing down context c is only conditioned on e , and estimate it using Eq. 3. This is a harsh approximation, which does not consider the impacts of relations r on c . As presented in the aforementioned examples, the relation r is often reflected by some words in c . If words w reflecting the relation r between e_f and e could be learnt from the data, they could potentially help rank all candidates. Intuitively, if words in c talk about r , candidates linked to e_f through r should receive more credits.

To model words reflecting a relation r , we assume that each relation $r \in R$ is associated with a multinomial distribution θ_r over the vocabulary of D . For example, $P(\text{better}|\theta_{\text{Co-type}})$ and $P(\text{than}|\theta_{\text{Co-type}})$ should be larger than $P(\text{food}|\theta_{\text{Co-type}})$. We also assume a multinomial “switch” variable s for each word occurrence, which takes three values “entity”, “relation”, and “other” with probabilities λ_1 , λ_2 , and $(1 - \lambda_1 - \lambda_2)$, respectively. When s takes one of the three values, the corresponding word w will be generated from $P(w|d_e)$, $P(w|\theta_r)$, or $P(w|D)$ accordingly. That is, we regard context c as a set of words independently generated from a mixture of $P(w|d_e)$, $P(w|\theta_r)$, and $P(w|D)$:

$$P(c|e_f, e, r) = \prod_{w \in c} [\lambda_1 P(w|d_e) + \lambda_2 P(w|\theta_r) + (1 - \lambda_1 - \lambda_2) P(w|D)] \quad (8)$$

Note that by adopting the above equation, the impacts of words related to e_f are included in $P(w|D)$. In Lines 3-12 of Algorithm 1, we sample w from such a mixture.

The parameters of FocalLink are $\{\pi_r\}_R$ for relations in Table 2, and the unknown word distributions $\{\theta_r\}_R$. We regard $\lambda = (\lambda_1, \lambda_2)$ as hyper-parameters. After hyper-parameters are specified and parameters are learnt, we link m according to

$$\begin{aligned} e^{(m)} &= \arg \max_{e \in \mathcal{E}(m)} P(e|c, e_f) \\ &= \arg \max_{e \in \mathcal{E}(m)} \sum_{r \in R} \pi_r P_G(e|e_f, r) P(c|e_f, e, r) \end{aligned} \quad (9)$$

Next, we discuss inferring the parameters of FocalLink.

4.3 Model Inference

Given a collection of unlabeled mentions $m = (e_f, a, c)$ with candidates, we adopt the **EM** algorithm to estimate parameters $\Phi = (\{\pi_r\}, \{\theta_r\})$.

In the **E-step**, given mentions m and the current estimation $\hat{\Phi} = (\{\hat{\pi}_r\}, \{\hat{\theta}_r\})$, we jointly treat e and r as hidden variables and compute the following posterior probability

$$P(e, r|m; \hat{\Phi}) = \frac{P(e, r, m; \hat{\Phi})}{\sum_{e', r'} P(e', r', m; \hat{\Phi})} \quad (10)$$

where $P(e, r, m; \hat{\Phi}) \propto P(c|e_f, e, r) P_G(e|e_f, r) \pi_r$.

Readers may notice that we do not involve hidden variables s in the E-step. In later derivations we will show that it is enough to compute the posterior at such granularity. For ease of notations, we will use S to denote the set of

Algorithm 2 EM-based parameter inference for FocalLink.

Input: Unlabeled mentions $\{m\}$ and candidates $\{\mathcal{E}(m)\}$

Output: Estimated parameters $\hat{\Phi}$, including prior probability $\{\hat{\pi}_r\}$ and word distribution $\{\hat{\theta}_r\}$ for relations $r \in R$

```

1: Randomly initialize  $\hat{\Phi} = (\{\hat{\pi}_r\}, \{\hat{\theta}_r\})$ ;
2: repeat
3:   for all unlabeled mention  $m$  do ▷ E-Step
4:     for all  $e \in \mathcal{E}(m), r \in R$  do
5:        $P_{e,r}^{(m)} \leftarrow P(c|e_f, e, r; \hat{\theta}_r) P_G(e|e_f, r) \hat{\pi}_r$ ;
6:     end for
7:     Normalize  $P^{(m)}$ ; ▷ Eq. 10
8:   end for
9:   for all  $r \in R$  do ▷ M-Step
10:     $\pi_r \leftarrow \frac{1}{|M|} \sum_m \sum_e P_{e,r}^{(m)}$ ; ▷ Eq. 11
11:    for all unlabeled mention  $m = \langle e_f, a, c \rangle$  do
12:      for all  $e \in \mathcal{E}(m), w \in c$  do
13:         $S_{e,w}^{(m)} \leftarrow P(S(w) = \text{rel}|m, e, r, \hat{\Phi})$ ; ▷ Eq. 12
14:      end for
15:    end for
16:    for all  $w$  in the vocabulary do
17:       $P_w^{(\theta_r)} \leftarrow \sum_m \sum_e \#(w, c) S_{e,w}^{(m)} P_{e,r}^{(m)}$ ;
18:    end for
19:    Normalize  $P^{(\theta_r)}$  as  $\theta_r$ ; ▷ Eq. 13
20:  end for
21:   $\hat{\Phi} \leftarrow (\{\pi_r\}, \{\theta_r\})$ ; ▷ Update current estimation
22: until convergence
23: return  $\hat{\Phi}$ ;

```

switch variables for all words in a context c , and $S(w)$ for the switch variable of word w .

Given the posterior in Eq. 10, the expectation of complete data log-likelihood is:

$$\begin{aligned} L(\Phi) &= \sum_m \sum_{e,r,S} P(e, r, S|m; \hat{\Phi}) \log P(e, r, S, m; \Phi) \\ &= \sum_m \sum_{e,r,S} \left[P(e, r, S|m; \hat{\Phi}) \right. \\ &\quad \left. \log P(c, S|e, r, e_f; \{\theta_r\}) P_G(e|e_f, r) P(r|e_f; \{\pi_r\}) P(e_f) \right] \end{aligned}$$

In the logarithm, $P_G(e|e_f, r)$ and $P(e_f)$ are not parameterized, thus not affecting the maximization of $L(\Phi)$. Therefore, maximizing the above likelihood is equivalent to maximizing the following two independent components:

$$\begin{aligned} G(\{\pi_r\}) &= \sum_m \sum_{e,r,S} P(e, r, S|m; \hat{\Phi}) \log P(r|e_f; \{\pi_r\}), \text{ and} \\ H(\{\theta_r\}) &= \sum_m \sum_{e,r,S} P(e, r, S|m; \hat{\Phi}) \log P(c, S|e, r, e_f; \{\theta_r\}) \end{aligned}$$

In the **M-step**, we seek to simplify G and H further before maximizing them. For G , we have

$$G(\{\pi_r\}) = \sum_m \sum_{e,r} P(e, r|m; \hat{\Phi}) \log \pi_r$$

Under constraint $\sum_r \pi_r = 1$, the maximizer is

$$\pi_r = \frac{1}{|M|} \sum_m \sum_e P(e, r|m; \hat{\Phi}) \quad (11)$$

For H , it could be seen as $|R|$ independent terms. We abuse H here to present and simplify one of them

$$\begin{aligned} H(\theta_r) &= \sum_m \sum_{e, S} P(e, r, S | m; \hat{\Phi}) \log P(c, S | e, r, e_f; \theta_r) \\ &= \sum_m \sum_{w \in c, S} \sum_e P(e, r, S | m; \hat{\Phi}) \log P(w, S(w) | e, r, e_f; \theta_r) \end{aligned}$$

Note that for $\forall w \in c$, the probability $P(w, S(w) | e, r, e_f; \theta_r)$ is a function of θ_r only when $S(w) = \text{rel}$ ("relation"). In this only case, $P(w, S(w) = \text{rel} | e, r, e_f; \theta_r) = \lambda_2 P(w | \theta_r)$. Therefore, maximizing $H(\theta_r)$ is equivalent to maximizing

$$\begin{aligned} &\sum_m \sum_{w \in c} \sum_e P(e, r, S(w) = \text{rel} | m; \hat{\Phi}) \log P(w | \theta_r) \\ &= \sum_m \sum_{w \in c} \sum_e P(S(w) = \text{rel} | m, e, r; \hat{\Phi}) P(e, r | m; \hat{\Phi}) \log P(w | \theta_r) \end{aligned}$$

Here

$$\begin{aligned} P(S(w) = \text{rel} | m, e, r; \hat{\Phi}) &= \\ &= \frac{\lambda_2 P(w | \hat{\theta}_r)}{\lambda_1 P(w | d_e) + \lambda_2 P(w | \hat{\theta}_r) + (1 - \lambda_1 - \lambda_2) P(w | D)} \end{aligned} \quad (12)$$

With respect to $\sum_w P(w | \theta_r) = 1$, the maximizer is

$$\begin{aligned} P(w | \theta_r) &= \\ &= \frac{\sum_m \sum_e \#(w, c) P(S(w) = \text{rel} | m, e, r; \hat{\Phi}) P(e, r | m; \hat{\Phi})}{\sum_m \sum_{w'} \sum_e \#(w', c) P(S(w') = \text{rel} | m, e, r; \hat{\Phi}) P(e, r | m; \hat{\Phi})} \end{aligned} \quad (13)$$

Here $\#(w, c)$ is the number of w 's occurrences in c .

The entire inference algorithm for FocalLink is presented in Algorithm 2. To summarize, the EM algorithm accepts a set of unlabeled mentions with candidates as input, and sequentially iterates over Eqs. 10, 11, 12, and 13 until convergence or a predefined number of rounds are completed.

5 COLLECTIVE LINKING

So far, all methods we have discussed are *local methods*, i.e., linking mentions independently. However, existing studies suggest that collectively linking a set of related mentions $m_i \in M$ leads to better linking accuracy [1], [30], [50]. The underlying assumption is that the ground truth entities e_i for related m_i should not only be compatible with information available in m_i , but also be *coherent*, i.e., geographically close to each other in our location linking setting. By encouraging coherence between linking decisions of mentions in M , positive impacts are expected on the linking results.

Given mentions $m_i \in M$ to be linked collectively and their candidate lists $\mathcal{E}(m_i)$, we link m_i to $e_i \in \mathcal{E}(m_i)$ by maximizing the following object function [50]:

$$\text{Obj} = \frac{\alpha}{|M|} \sum_i P(e_i | m_i) + \frac{1 - \alpha}{\binom{|M|}{2}} \sum_{i < j} n\text{Dist}(e_i, e_j), \quad (14)$$

where $P(e_i | m_i)$ is estimated by an arbitrary local method. $n\text{Dist}(e_i, e_j)$ is the normalized distance between linking decisions e_i and e_j , which is defined in Eq. 6. α is a parameter balancing the averaged local linking probability and averaged pairwise distance.

Finding the exact solution to the above optimization problem is NP-HARD [50]. Therefore, we adopt a greedy algorithm named *iterative substitution algorithm* [50]. First, we start with an initial solution where each e_i is chosen such that $P(e_i | m_i)$ is maximized. Then we carry out the following iteration procedure until convergence. In each iteration, we scan the linking decision e_i of each mention m_i , and attempt to change it to another decision e'_i . After the scan, the decision change which can lead to the largest improvement of the objective function is applied. According to [50], the objective function is guaranteed to converge.

Due to the speciality of our problem, to instantiate mentions M being linked collectively, we have two choices:

- *Comment level.* M consists of all mentions detected from the comment $\mathcal{C}$.
- *Focal-entity level.* M consists of not only all mentions in $\mathcal{C}$, but also *some* mentions detected from other comments on the profile page of e_f . For further reference, we call the latter kind of mentions *background mentions*. To study the impacts of different number of background mentions on linking accuracy, we make the number of background mentions a parameter #BgMen, and vary it in the experimental study.

We study the above two variations of collective linking because neither of them is intuitively superior to the other. On the one hand, comment-level collective linking requires that the user comment contains at least two mentions. However, according to a data sample of 4,000 user comments where 648 contains at least one location mention, only 137 (21.1%) contains two or more mentions. In other words, for every five comments with mentions, only one can benefit from comment-level collective linking. On the other hand, focal-entity-level collective linking does not suffer from this disadvantage because there are potentially many background mentions available. However, it is not clear whether background mentions will bring accuracy improvements like other mentions in the same comment. We will investigate the effectiveness of the two settings in Section 6.2.

6 EXPERIMENTS

6.1 Experimental Preparations

In this subsection, we report experimental preparations prior to linking. First, we describe the collected Foursquare dataset. Recall that both the training of FocalLink and focal-entity-level collective linking require an NER classifier to detect mentions. We describe here the features used to train the NER classifier and report its performance. Finally, as all models share the same candidate generation component (see Section 3.2.1), we report the results of candidate generation.

6.1.1 Dataset

Constructing the Location Graph. We collected 321,943 locations located in Singapore and their 442,803 comments in English, through Foursquare API. The attributes of each location include its full name, description, categories, parent (location), coordinates, and number of historical check-ins.

TABLE 3
Summary of the Foursquare dataset.

Item	Count
# Locations	321,943
# Type relation instances	276,620
# <i>inside</i> edges before completion	4,979
# <i>inside</i> edges after completion	55,813
# Comments	442,803
# Sampled comments	4,000
# Labeled location mentions	828
# Mentions linked to NULL	115
# Comments with at least one location mention	648
# Comments with at least two location mentions	137

TABLE 5
Rules used to complete the *inside* edges.

Rule	Example
e_1 's address contains at least two words from e_2 's name.	e_1 .address = 1 Maritime Square #01-18/19 HarbourFront Centre e_2 .name = HarborFront Centre
e_1 's name contains '@', followed by e_2 's name or abbreviation.	e_1 .name = Amberrock@USS e_2 .name = Universal Studios Singapore

Edges in the data graph $\mathcal{G}$ depend on those attributes as follows:

- The *inside* edges are built with the “parent” attribute of locations. Because this attribute is crowd-sourced from users, and we found that very few locations contain such information. Therefore, we used two rules to complete the *inside* edges. A location e_1 is considered to be inside e_2 if they meet the following two conditions: (i) e_1 's full address (or its name) matches e_2 's name or it's abbreviation (see Table 5 for examples); and (ii) they are geographically close to each other. After the completion, the number of *inside* edges increased from 4,979 to 55,813.
- The *type* edges are built on the “categories” attributes. We include one node for each possible type, and link the node of a location to all its type nodes.
- The distance edges are computed with geographical coordinates (latitudes and longitudes). They are not pre-computed, but are only computed in Eq. 6 when necessary.

Finally, we note that instances of relations *Co-Inside* and *Co-Type* in Table 2 are also not materialized before-hand. They are only implicitly traversed along, but not stored, when computing random walk probabilities in Eq. 7. Table 3 reports the summary of the data used in our experiments.

Collecting and Labelling Comments. We uniformly sampled 4,000 comments from the 442,803 comments. They were evenly assigned to two human annotators who are familiar with Singapore locations. The annotators were asked to label possible location anchors using the BIO labeling scheme (*i.e.*, the **B**eginning, **I**nside, and **O**utside of an anchor), and find out which location this anchor refers to. When the annotators could not decide the true location of an anchor, a

TABLE 4
Summary of anchor recognition features.

Basic features
The word itself and lowercased form.
Prefix and suffix of the current word, with length up to 3.
Word shape: capitalized, all capitalized, all numeric, alphanumeric.
Preceding and following two words in lowercased form.
Bag-of-words in the context window of length 5.
User language specific features
Brown clustering [51] features based on path prefix of length 4, 8, and 12.
Dictionary-based features: BIO <i>pre-labels</i> [52], [53] of AML [54] match and Trie match.

NULL mark was given. The annotators were encouraged to refer to search engines to assist their labeling work. After they both finished their own parts, they cross-checked the other one's part and reached agreements. Finally, 828 mentions were found, 115 out of which were marked as NULL, as shown in Table 3.

6.1.2 Location Recognition Performance

To train FocalLink unsupervisingly, potential anchors in unlabeled comments need to be recognized to produce unlabeled training mentions. Moreover, as introduced in Section 5, background mentions should be prepared for focal-entity-level collective linking. Conditional Random Field (CRF, [47]) becomes a natural choice for fulfilling this task. We note that there are publicly available off-the-shelf NER tools, *e.g.*, StanfordNER [55] and TwitterNLP [7]. However, we found that their performance remains inferior on our data. Therefore, we decided to train our own NER model.

Our NER model is trained on the labeled data with the CRF++⁵ toolkit with default parameters. The features we used are described in Table 4. Besides the commonly used NER features, we also included the following two types of features to deal with the noise in comments.

Word group by Brown clustering. The Brown clustering algorithm [51] has proved effective in finding misspelling, transliteration, or informal abbreviations of words. For example, for the phrase “kway teow” (ricecake strips), Brown clustering can identify different forms of “kway” like “kwey”, “kuay”, and “kuey”. For a given text corpus, Brown clustering outputs a hierarchy of clusters for all words based on their contextual similarity. Following [7], [53], [56], for each word, we used its hierarchy path prefixes of length 4, 8, and 12 as features.

Dictionary features. The location names in the database $\mathcal{E}$ could be viewed as a dictionary and used to extract anchors from comments by exact matching. However, it will suffer from low recall, because users tend to use varied or abbreviated location names instead of full names in comments. To deal with such issues, we perform Approximate Member Localization (AML) [54] to identify anchors similar to full names, *e.g.*, *Charles n Keith* for Charles & Keith, and Trie-based longest prefix match (Trie) for anchors that are prefixes of full names, *e.g.*, *Brotzeit* for Brotzeit German

5. <https://taku910.github.io/crfpp/>

TABLE 6
Anchor recognition performance.

Features	Prec	Rec	F ₁
StanfordNER (Retrained)	.726	.354	.473
TwitterNLP (Non-retrainable)	.548	.214	.306
All - BrownClustering	.731	.512	.599
All - AML	.746	.540	.625
All - Trie	.722	.542	.618
All - AML - Trie	.698	.480	.568
All Features	.755	.570	.647

Bier Bar & Restaurant. Following [52], [53], we transform the output of these two operations into two separate pre-label features. For each feature, BIO labels are used to mark potential anchors identified by the corresponding matching operation. CRF model will consider information from other features together with the pre-labels to produce final predictions.

In Table 6, we analyze the performance of our NER model under 10-fold cross validation. We also present the best-effort results of StanfordNER and TwitterNLP. We re-trained StanfordNER on our dataset with default features and parameters. To make a fair comparison, the full names of all locations were also provided to StanfordNER as a gazetteer. Sloppy matching is enabled to capture partial location names in users’ comments. However, the recall of StanfordNER is as low as 0.354. We note that, compared with Brown clustering and AML, the sloppy match of StanfordNER cannot well handle cases like misspelling and word reordering in informal user language. On the other hand, the performance of TwitterNLP is even worse than that of StanfordNER. One reason is that the model provided by TwitterNLP is fixed, *i.e.*, not retrainable. Another possible explanation is that TwitterNLP is less effective in recognizing fine-grained location entities (with an F₁ score of 0.37), compared with person or company entities (F₁ scores are 0.82 and 0.71, respectively) [7].

To validate the necessities of features handling informal user language, we also conducted ablations studies. When Brown clustering is removed, the recall significantly decreases from 0.57 to 0.512. This is because its absence may reduce the model’s generalization power from a word to its potential variations. As for the two pre-label features based on AML and Trie, removing any of them causes all three metrics to drop slightly. However, the model suffers from poor performance without both of them. This result indicates that the two features are correlated with but also slightly complement to each other.

To support the training of FocalLink and focal-entity-level collective linking, we train the above NER classifier on the labeled data, and apply it to the unlabeled data. Among the 438,803 unlabeled comments, our NER classifier found 64,198 mentions. There are 51,976 (11.8%) comments with at least one mention, and 8,932 with at least two.

6.1.3 Candidate Generation Performance

Using the labeled data, we quantify the impacts of the four rules on candidate generation introduced in Section 3.2.1 by Table 7. The impact of each rule is characterized by

TABLE 7
Impacts of candidate matching rules.

Rule	Used solely	Absent
Jaccard	300 (42.1%)	588 (82.5%)
Prefix	459 (64.4%)	553 (77.6%)
WordCover	508 (71.2%)	528 (74.1%)
Acronym	47 (6.6%)	563 (79.0%)
Mens. covered when all used	597 (83.7%)	
Non-NULL mentions	713	

the number and percentage of non-NULL mentions whose ground truth location can be retrieved, when the rule is used solely or absent from the four rules. Throughout this study, we also empirically set the Jaccard similarity threshold τ to 0.7. Table 7 shows that the rule of word cover (*i.e.*, the words of the anchor should be covered by the full name of a candidate) can retrieve the most number of ground truth locations, *i.e.*, 71.2% of non-NULL mentions. However, 89 mentions (12.5%) still needs to be addressed by the other three rules. For labeled mentions, the average and median numbers of generated candidates are 122 and 11, respectively. After recognizing all anchors in unlabeled comments, we also apply candidate generation on them. This results in 32,617 background mentions with at least one candidate, which are responsible for training FocalLink and applying focal-entity-level collective linking.

6.2 Experimental Results for Linking

In this subsection, we demonstrate the linking performance of FocalLink and the aforementioned baselines. First, we report the settings adopted by all linking methods. Then we compare all methods under three settings w.r.t. whether and how collective linking (CL) is applied, *i.e.*, non-CL, comment-level CL, and focal-entity-level CL. In those comparisons, we also remove the *Near* relation in FocalLink to study its potential of generalizing to other non-geographical domains. For the remainder of this subsection, we use concrete examples to qualitatively make sense of FocalLink.

6.2.1 Miscellaneous Linking Settings

In experiments, we use precision (Prec), recall (Rec), and F₁ to evaluate location linking results. Let S be the system output, and S^* be the ground truth. Precision, recall, and F₁ are defined as following:

$$Prec = \frac{|S \cap S^*|}{|S|}, Rec = \frac{|S \cap S^*|}{|S^*|}, F_1 = \frac{2 * Prec * Rec}{Prec + Rec}.$$

We use all manually labeled mentions to evaluate FocalLink and the three baselines in Section 3.2.2, where the labeled anchors are provided to them as input. Parameters in those methods are empirically set as follows: $\lambda_1 = 0.2$, $\lambda_2 = 0.1$, and $Maxdist = 3km$. When training FocalLink with the EM algorithm, we iterate for 10 times. Besides testing all methods in a mention-by-mention, non-collective-linking manner, we also compare them under the two collective linking settings introduced in Section 5. All parameters and settings in the non-CL experiments are adopted in the collective linking experiments.

TABLE 8
Linking result summary of all methods w.r.t. different collective linking (CL) settings.

Approaches	Non-CL			Comment-lvl CL				Focal-entity-lvl CL			
	Prec	Rec	F ₁	Prec	Rec	F ₁	Opt. α	Prec	Rec	F ₁	Opt. #BgMen
Popularity (Eq. 1)	.563	.508	.534	.584	.527	.554	0.5	.616	.555	.584	10
PopContext (Eq. 2)	.619	.558	.587	.636	.573	.603	0.7	.669	.603	.634	16
PopContextDist (Eq. 4)	.679	.611	.643	.679	.611	.643	0.7	.682	.615	.646	1
FocalLink (w/o Near)	.648	.584	.614	.655	.590	.620	0.6	.661	.596	.627	9
FocalLink (Eq. 9)	.690	.621	.653	.693	.624	.657	0.6	.696	.627	.660	3

Fig. 4. Performance details of all methods under different collective linking settings.

6.2.2 Quantitative Analysis

In Table 8, we summarize the performance of all methods w.r.t. the aforementioned collective linking settings.

Non-CL. In the non-CL columns, the performance of all baselines and the full version of FocalLink increases in the order they are presented. The PopContext baseline outperforms Popularity by 5 points in terms of all metrics. This is because the compatibility between related words of a candidate and the context words may imply that the comment is mentioning the candidate. When the distance between a candidate and the focal location is taken into consideration, PopContextDist again improves all three metrics by at least 5 points. This suggests that, for most of the time, users tend to use the focal location as a geographical context and mention nearby locations.

After involving relations except *Near*, the performance of FocalLink (w/o *Near*) is sandwiched between PopContext and PopContextDist. The result that PopContextDist outperforms FocalLink (w/o *Near*) shows the importance of distance signal in the location domain. However, we note that the superiority of FocalLink (w/o *Near*) to PopContext demonstrates the potential of generalizing FocalLink to other domains, where distance information is not available. Finally, after taking the *Near* relation into consideration, the full version of FocalLink manages to achieve one point of improvement over PopContextDist, the most competitive baseline. Readers may notice that the Prec/Recall/F₁ of all methods does not exceed 0.7. This is because their performance not only depends on candidate ranking, but also on the shared candidate generation component. In Section 6.1.3, we report that this component recalls ground-truth entities for 83.7% of non-NULLE mentions. This means any linking model on this dataset cannot achieve Recall scores higher than 0.837. Moreover, for the 16.3% of mentions whose ground-truths cannot be retrieved, if false candidates are

retrieved, the Precision score will be degraded. Candidate generation for comments is much harder than for formal texts because of their informal writing style and various abbreviations. Although currently not our key focus, its further improvement would help relief this problem. In Figure 4(a), we show the F₁ score of FocalLink when it is fed with different number of training mentions. It is observable that with a small amount of unlabeled data, FocalLink is able to overtake PopContextDist. As more data is provided, FocalLink can still benefit from the training process.

Comment-level CL. In Figure 4(b), we adopt all configurations from the non-CL setting, and show F₁ scores of the four methods with varying α under the comment-level CL setting. This figure demonstrates that α generally controls the tradeoff between local linking probability and coherence in Eq. 14. The best performance and optimal α 's are reported in the middle columns of Table 8. We observe that comment-level-CL approaches generally bring slight or even no improvement. Specially, the comment-level-CL versions of all baselines cannot even beat the non-CL version of FocalLink. Recall in Section 5, we mentioned that, due to the short length of comments, only 21.1% of them have two or more mentions. In other words, most comments cannot benefit from comment-level collective linking. Despite the slight improvements, the comment-level-CL version of FocalLink outperforms all baselines.

Focal-entity-level CL. In Figure 4(c), we adopt all optimal configurations from the comment-level CL setting, and show F₁ scores of focal-entity-level CL approaches when different #BgMen are used in collective linking. For Popularity and PopContext, as more background mentions are added, the F₁ scores generally fluctuate while following an increasing trend, and the trend converges when #BgMen exceeds 20. To explain this, in the same figure, we show

TABLE 9
Impacts and statistics of relations.

Relation	Prec	Rec	F ₁	Count
<i>Self-Ref</i>	.568	.512	.539	150
<i>Inside</i>	.566	.510	.537	32
<i>Contains</i>	.579	.522	.549	167
<i>Co-Inside</i>	.572	.516	.542	45
<i>Co-Type</i>	.548	.494	.519	121
<i>Near</i>	.685	.617	.649	498
All	.690	.621	.653	828

TABLE 10
Relation words $\{\theta_r\}$ learnt on all unlabeled data.

<i>Self-Ref</i>		<i>Inside</i>		<i>Contains</i>		<i>Co-Inside</i>		<i>Co-Type</i>		<i>Near</i>	
at	.068	st	.021	at	.030	plaza	.022	better	.059	to	.038
to	.048	at	.013	stall	.026	thomson	.021	than	.057	from	.030
in	.047	cross	.010	noodle	.017	at	.019	at	.029	at	.025
on	.047	outlet	.010	on	.013	go	.019	on	.019	bus	.021
the	.031	locate	.010	from	.010	hungry	.012	much	.017	walk	.014
it	.030	rd	.008	must	.009	atm	.011	compare	.015	locate	.013
of	.024	open	.008	fry	.009	check	.011	cheaper	.015	road	.013
s	.020	on	.008	mee	.009	out	.009	as	.014	mrt	.013
is	.020	road	.008	serve	.008	beside	.009	to	.013	take	.012
from	.019	middle	.008	has	.007	wordpress	.009	price	.012	go	.010

the number of labeled mentions which are not linked to NULL and have at least #BgMen background mentions. For example, among all 713 non-NULL mentions, less than 300 of them have more than 30 background mentions. For larger #BgMen, fewer mentions benefit from focal-entity-level CL. This explains why the F₁ scores stop increasing. Observe that FocalLink (w/o *Near*) performs better than PopContext until #BgMen reaches 10. We note that, when *Near* edges are not considered, the data graph becomes very sparse. This may cause bad cases in FocalLink (w/o *Near*) on some background mentions, which then misleads collective linking. We also observe that the lines of PopContextDist and FocalLink even stop growing at very small #BgMen values (1 for PopContextDist and 3 for FocalLink). This is because the two methods have access to the geographical coordinates of the focal location, which is more informative than background mentions. Again, FocalLink remains competitive in this setting. The Focal-entity-level CL versions of all baselines with optimal numbers of background mentions are even poorer than the non-CL version of FocalLink.

Analysis on Relations. In Table 9, we analyze the contribution of each relation to FocalLink. We also include the statistics in Table 2 on how many comments each relation is involved in. We adopt the non-CL setting and compare the performance when each relation is solely used. By comparing this table and the performance of non-CL FocalLink in Table 8, we make the following observations. First, when all relations are used, the performance is superior to that when only one of them is used, in terms of all metrics. Second, the relation *Near* has the best stand-alone performance among all six relations. This is in consistency with the statistics. Although we performed rule-based completion on the graph, the spatial containment and type information may still be very sparse compared to *Near*. Finally, the performance of *Near* relation is slightly better than that of non-CL PopContextDist in Table 8. The reason is that FocalLink models words related to a relation. When words indicative of *Near* are observed in the surrounding context, $P(w|e_f, e, r)$ in FocalLink will be less affected by $P(w|d_e)$, making it more concentrated on the distance information in $P(e|e_f, \text{Near})$.

6.2.3 Qualitative Analysis

Making Sense of Relation Words. We now compare relation-indicating words learnt by FocalLink. Table 10

presents top-10 most probable w w.r.t. $P(w|\theta_r)$ for each relation r , together with their probability. We interpret them below.

1) Indicative words for *Self-Ref* are mostly prepositions. From the data we observe that if the focal entity e_f itself is mentioned in a comment, they are usually modified by those indicative words. Besides the example “a visit to Zouk is ...” in Table 2, another example is $\langle \text{Shinji by Kanesaka (a Japanese restaurant), “The other day at Shinji ...”} \rangle$.

2) For relation *Inside*, words like “at”, “cross”, “locate”, “on” and “middle” modify the mentioned location where e_f is inside, e.g., $\langle \text{Crowne Plaza Changi Airport (an airport hotel), “Conveniently located in terminal 1 ...”} \rangle$. Meanwhile, “st”, “rd”, and “road” are used to give further directions.

3) For relation *Contains*, we find that this relation is usually involved when a user leaves comments on a mall or food court e_f . Those comments usually point directly to a restaurant or food stall e contained by e_f with a recommending tongue (e.g., starting with “**Must** try e ”).

4) For *Co-Inside*, because we have fewer comments involving this relation to learn a good word distribution (readers may refer to the last column of Table 9 though we actually learn the words on unlabeled data), the words are not as intuitive. Some words are related to specific locations, e.g., “plaza” and “thomson”. However, words like “check out” and “beside” still convey this relation. Moreover, “wordpress” is caused by spamming merchants posting ads on the page of their competitors in the same mall. Those ads contain URLs of websites hosted on wordpress.com.

5) For *Co-Type*, readers may notice that almost all words here are related to comparisons. As indicated by the probability, the most common comments for comparisons are like $\langle e_f, \text{“Better than } e \text{.”} \rangle$. Besides, users often write comparative degrees of adjectives following the word “much”, though most of the time they care for “much cheaper price”.

6) For *Near*, most words are distance-related prepositions (“to” and “from”), verbs (“take” and “go”), and transportation means (“bus”, “walk”, and “mrt”). This suggests that users’ comments here tend to describe travel options between the focal and mentioned entities, e.g., $\langle \text{Bus Stop 08031, “Take 77 for a shorter trip to bukit timah plaza ...”} \rangle$.

A Case Study. In Table 11, we demonstrate a case where FocalLink wins the other three baselines. The comment says “Better than MBS” and is posted on the page of Royal Plaza On Scotts, a hotel in Singapore. Note that

TABLE 11
Linking results for “MBS” in (Royal Plaza On Scotts, “Better than MBS.”).

Rank	Popularity & PopContext	PopContextDist	FocalLink
1	Marina Bay Sands	Marina Bay Sands	Marina Bay Sands Hotel
2	Marina Bay Sands Hotel	Marina Bay Sands Hotel	Tower 1 Marina Bay Sands Hotel
3	The Shoppes At Marina Bay Sands	The Shoppes At Marina Bay Sands	Tower 3 Marina Bay Sands Hotel
4	Marina Bay Sands Casino	Marina Bay Sands Casino	Tower 2 Marina Bay Sands Hotel
5	Marina Bay Sands Boardwalk	Marina Bay Sands Boardwalk	The Club At Marina Bay Sands Singapore
6	Tower 1 Marina Bay Sands Hotel	Tower 3 Marina Bay Sands Hotel	Marina Bay Sands Atrium 1
7	Tower 3 Marina Bay Sands Hotel	Tower 1 Marina Bay Sands Hotel	Marina Bay Sands
8	Tower 2 Marina Bay Sands Hotel	Marina Bay Street Circuit	Marina Bay Sands Butler Office Level 54
9	Marina Bay Street Circuit	Tower 2 Marina Bay Sands Hotel	The Shoppes At Marina Bay Sands
10	Marina Bay Sands Team Dining Room (TDR)	Macpherson BBQ Seafood	Marina Bay Sand Tower Bar

“MBS” is an acronym of “Marina Bay Sands”, a famous integrated resort in Singapore. It includes various affiliated locations, *e.g.*, a hotel, a casino, a mall, a conventional center, *etc.*, which are co-located and have individual entries in Foursquare database. Therefore, the most appropriate linking choice of the “MBS” here would be *Marina Bay Sands Hotel*, since the user aims at comparing the two hotels. From the table we see that the Popularity method fails to rank the hotel at top-1, because the general Marina Bay Sands has more check-ins than the hotel. The PopContext method ends up with the same results, because there is rarely any hotel-related keywords to leverage in the short surrounding context. Although knowing the focal location, the PopContextDist method could make little difference, because all top candidates have approximately equal distance from the focal location. Finally, by identifying the *Co-Type* relation indicated by “better than” and referring to the data graph, *FocalLink* gives more credit to hotel-typed candidates and is able to catch the right answer.

7 CONCLUSION

User comments are a major type of user generated content. In this paper, we address linking locations mentioned in Foursquare comments, which may improve tasks like comment gathering, sentimental analysis, and location recommendation. The proposed solution deals with the short characteristics of user comments. To deal with the insufficiency of context for disambiguating mentions, we exploit the focal location and the relations between locations as extra contextual information. More importantly, the data graph enables estimating the possibility that a user mentions a location while commenting on another. Our model incorporates all these cues that may help location linking. Experiments show that our solution achieves superior performance over three baseline methods under different collective linking settings.

ACKNOWLEDGMENT

This work was supported by Singapore Ministry of Education Research Fund MOE2014-T2-2-066. Wayne Xin Zhao was partially supported by National Natural Science Foundation of China under Grant No. 61502502 and Beijing Natural Science Foundation under Grant No. 4162032.

REFERENCES

- [1] S. Cucerzan, “Large-scale named entity disambiguation based on wikipedia data,” in *EMNLP-CoNLL*, vol. 7, 2007, pp. 708–716.

- [2] D. E. Appelt, J. R. Hobbs, J. Bear, D. Israel, and M. Tyson, “Fastus: A finite-state processor for information extraction from real-world text,” in *IJCAI*, vol. 93, 1993, pp. 1172–1178.
- [3] L. Ratnov and D. Roth, “Design challenges and misconceptions in named entity recognition,” in *CoNLL*, 2009, pp. 147–155.
- [4] R. Mihalcea and A. Csomai, “Wikify!: linking documents to encyclopedic knowledge,” in *CIKM*, 2007, pp. 233–242.
- [5] D. Nadeau and S. Sekine, “A survey of named entity recognition and classification,” *Linguisticae Investigationes*, vol. 30, no. 1, pp. 3–26, 2007.
- [6] W. Shen, J. Wang, and J. Han, “Entity linking with a knowledge base: Issues, techniques, and solutions,” *TKDE*, vol. 27, no. 2, pp. 443–460, 2015.
- [7] A. Ritter, S. Clark, O. Etzioni *et al.*, “Named entity recognition in tweets: an experimental study,” in *EMNLP*, 2011, pp. 1524–1534.
- [8] X. Liu, F. Wei, S. Zhang, and M. Zhou, “Named entity recognition for tweets,” *TIST*, vol. 4, no. 1, p. 3, 2013.
- [9] C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee, “Twinner: named entity recognition in targeted twitter stream,” in *SIGIR*, 2012, pp. 721–730.
- [10] Y. Fang and M.-W. Chang, “Entity linking on microblogs with spatial and temporal signals,” *TACL*, vol. 2, pp. 259–272, 2014.
- [11] W. Hua, K. Zheng, and X. Zhou, “Microblog entity linking with social temporal context,” in *SIGMOD*, 2015, pp. 1761–1775.
- [12] W. Shen, J. Wang, P. Luo, and M. Wang, “Linking named entities in tweets with knowledge base via user interest modeling,” in *SIGKDD*, 2013, pp. 68–76.
- [13] A. Davis, A. Veloso, A. S. Da Silva, W. Meira Jr, and A. H. Laender, “Named entity disambiguation in streaming data,” in *ACL*, 2012, pp. 815–824.
- [14] X. Liu, Y. Li, H. Wu, M. Zhou, F. Wei, and Y. Lu, “Entity linking for tweets,” in *ACL*, 2013, pp. 1304–1311.
- [15] Z. Ji, A. Sun, G. Cong, and J. Han, “Joint recognition and linking of fine-grained locations from tweets,” in *WWW*, 2016, pp. 1271–1281.
- [16] X. Liu, M. Zhou, F. Wei, Z. Fu, and X. Zhou, “Joint inference of named entity recognition and normalization for tweets,” in *ACL*, 2012, pp. 526–535.
- [17] G. Luo, X. Huang, C.-Y. Lin, and Z. Nie, “Joint named entity recognition and disambiguation,” in *ACL-IJCNLP*, 2015, pp. 879–888.
- [18] S. Guo, M.-W. Chang, and E. Kiciman, “To link or not to link? a study on end-to-end tweet entity linking,” in *HLT-NAACL*, 2013, pp. 1020–1030.
- [19] A. Sil and A. Yates, “Re-ranking for joint named-entity recognition and linking,” in *CIKM*, 2013, pp. 2369–2374.
- [20] B. Pang, L. Lee *et al.*, “Opinion mining and sentiment analysis,” *Foundations and Trends® in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.
- [21] B. Liu, “Sentiment analysis and opinion mining,” *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.
- [22] X. Ren, A. El-Kishky, C. Wang, F. Tao, C. R. Voss, and J. Han, “Clustype: Effective entity recognition and typing by relation phrase-based clustering,” in *SIGKDD*, 2015, pp. 995–1004.
- [23] H. Xu, S. Xie, L. Shu, and P. S. Yu, “Cer: Complementary entity recognition via knowledge expansion on large unlabeled product reviews,” in *BigData*, 2016, pp. 793–802.
- [24] F. Hasibi, K. Balog, and S. E. Bratsberg, “Entity linking in queries: Tasks and evaluation,” in *ICTIR*, 2015, pp. 171–180.

- [25] R. Blanco, G. Ottaviano, and E. Meij, "Fast and space-efficient entity linking for queries," in *WSDM*, 2015, pp. 179–188.
- [26] P. Deepak, S. Ranu, P. Banerjee, and S. Mehta, "Entity linking for web search queries," in *ECIR*, 2015, pp. 394–399.
- [27] F. Hasibi, K. Balog, and S. E. Bratsberg, "Exploiting entity linking in queries for entity retrieval," in *ICTIR*, 2016, pp. 209–218.
- [28] X. Han and L. Sun, "A generative entity-mention model for linking entities with knowledge base," in *ACL-HLT*, 2011, pp. 945–954.
- [29] E. Meij, W. Weerkamp, and M. de Rijke, "Adding semantics to microblog posts," in *WSDM*, 2012, pp. 563–572.
- [30] W. Shen, J. Wang, P. Luo, and M. Wang, "Linden: linking named entities with knowledge base via semantic knowledge," in *WWW*, 2012, pp. 449–458.
- [31] O.-E. Ganea, M. Ganea, A. Lucchi, C. Eickhoff, and T. Hofmann, "Probabilistic bag-of-hyperlinks model for entity linking," in *WWW*, 2016, pp. 927–938.
- [32] W. Shen, J. Han, and J. Wang, "A probabilistic model for linking named entities in web text with heterogeneous information networks," in *SIGMOD*, 2014, pp. 1199–1210.
- [33] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Vldb*, pp. 992–1003, 2011.
- [34] B. Sundheim, "Overview of results of the MUC-6 evaluation," in *Proceedings of the 6th Conference on Message Understanding*, 1995.
- [35] N. Chinchor, "Overview of MUC-7/MET-2," in *Proceedings of the 7th Conference on Message Understanding*, 1997.
- [36] G. R. Doddington, A. Mitchell, M. A. Przybicki, L. A. Ramshaw, S. Strassel, and R. M. Weischedel, "The automatic content extraction (ACE) program - tasks, data, and evaluation," in *Proceedings of the Fourth International Conference on Language Resources and Evaluation*, vol. 2, 2004, pp. 837–840.
- [37] I. Mani, C. Doran, D. Harris, J. Hitzeman, R. Quimby, J. Richer, B. Wellner, S. A. Mardis, and S. Clancy, "Spatialml: annotation scheme, resources, and evaluation," *Language Resources and Evaluation*, vol. 44, no. 3, pp. 263–280, 2010.
- [38] J. A. Bateman, J. Hois, R. J. Ross, and T. Tenbrink, "A linguistic ontology of space for natural language processing," *Artificial Intelligence*, vol. 174, no. 14, pp. 1027–1071, 2010.
- [39] L. Talmy, "Figure and ground in complex sentences," in *Annual Meeting of the Berkeley Linguistics Society*, vol. 1, 1975, pp. 419–430.
- [40] P. Kordjamshidi, M. van Otterlo, and M. Moens, "Spatial role labeling: Towards extraction of spatial relations from natural language," *TSLP*, vol. 8, no. 3, pp. 4:1–4:36, 2011.
- [41] M. Cornolti, P. Ferragina, and M. Ciaramita, "A framework for benchmarking entity-annotation systems," in *WWW*, 2013, pp. 249–260.
- [42] R. Usbeck, M. Röder, A.-C. Ngonga Ngomo, C. Baron, A. Both, M. Brümmer, D. Ceccarelli, M. Cornolti, D. Cherix, B. Eickmann *et al.*, "Gerbil: general entity annotator benchmarking framework," in *WWW*, 2015, pp. 1133–1143.
- [43] J. L. Leidner, *Toponym resolution in text: Annotation, evaluation and applications of spatial grounding of place names*. Universal-Publishers, 2008.
- [44] M. D. Lieberman and H. Samet, "Adaptive context features for toponym resolution in streaming news," in *SIGIR*, 2012, pp. 731–740.
- [45] M. D. Adelfio and H. Samet, "Geowhiz: toponym resolution using common categories," in *SIGSPATIAL*, 2013, pp. 532–535.
- [46] N. Dalvi, R. Kumar, and B. Pang, "Object matching in tweets with spatial models," in *WSDM*, 2012, pp. 43–52.
- [47] J. Lafferty, A. McCallum, F. Pereira *et al.*, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML*, 2001, pp. 282–289.
- [48] J. M. Ponte and W. B. Croft, "A language modeling approach to information retrieval," in *SIGIR*, 1998, pp. 275–281.
- [49] N. Lao and W. W. Cohen, "Relational retrieval using a combination of path-constrained random walks," *Machine Learning*, vol. 81, no. 1, pp. 53–67, 2010.
- [50] W. Shen, J. Wang, P. Luo, and M. Wang, "Liege:: link entities in web lists with knowledge base," in *SIGKDD*, 2012, pp. 1424–1432.
- [51] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, "Class-based n-gram models of natural language," *Computational linguistics*, vol. 18, no. 4, pp. 467–479, 1992.
- [52] J. Kazama and K. Torisawa, "Exploiting wikipedia as external knowledge for named entity recognition," in *EMNLP-CoNLL*, 2007, pp. 698–707.
- [53] C. Li and A. Sun, "Fine-grained location extraction from tweets with temporal awareness," in *SIGIR*, 2014, pp. 43–52.
- [54] Z. Li, L. Sitbon, L. Wang, X. Zhou, and X. Du, "Aml: Efficient approximate membership localization within a web-based join framework," *TKDE*, vol. 25, no. 2, pp. 298–310, 2013.
- [55] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *ACL*, 2005, pp. 363–370.
- [56] X. Ling and D. S. Weld, "Fine-grained entity recognition," in *AAAI*, 2012, pp. 94–100.

Jialong Han is a postdoctoral research fellow at School of Computer Science and Engineering, Nanyang Technological University. He earned his Ph.D. degree from Renmin University of China in 2015, under the supervision of Prof. Ji-Rong Wen. He obtained his B.E. degree also from Renmin University of China in 2010. His research interests include graph data mining and management, as well as their applications on knowledge graphs.

Aixin Sun is an Associate Professor with School of Computer Science and Engineering, Nanyang Technological University, Singapore. He received PhD from the same school in 2004. His research interests include information retrieval, text mining, social computing, and multimedia. His papers appear in major international conferences like SIGIR, KDD, WSDM, ACM Multimedia, and journals including DMKD, TKDE, and JASIST.

Gao Cong received the PhD degree from the National University of Singapore, in 2004. He is an associate professor with Nanyang Technological University, Singapore. Before he relocated to Singapore, he worked with Aalborg University, Microsoft Research Asia, and the University of Edinburgh. His current research interests include geo-textual data management and data mining.

Wayne Xin Zhao received the PhD degree from Peking University in 2014. He is currently an associate professor in the School of Information, Renmin University of China. His research interests are web text mining and natural language processing. He has published several referred papers in international conferences and journals such as ACL, EMNLP, COLING, ECIR, CIKM, SIGIR, SIGKDD, AAAI, IJCAI, ACM TOIS, ACM TKDD, ACM TIST and IEEE TKDE.

Zongcheng Ji received the Ph.D. degree from Institute of Computing Technology, Chinese Academy of Sciences. He has been a research fellow at Nanyang Technological University, Singapore. He is currently a postdoctoral research fellow at the University of Texas. His research interests include information retrieval, information extraction, and natural language processing.

Minh C. Phan is a Ph.D candidate at School of Computer Science and Engineering, Nanyang Technological University, under the supervision of Assoc. Prof. Sun Aixin. He received the B.E. degree in Computer Science from the same university in 2015. His research interests include information retrieval, text mining, entity resolution, and entity linking.