I.R. Lab
@shandong univ.

Social Media as Sensor in Real World: Geolocate User with Microblog

XueqinSui, Zhumin Chen, KaiWu, JunMa,
PengjieRen and FengyuZhou

Information Retrieval Lab.

Shandong University

OUTLINE

- **Problem Background**
- **Related Work**
- **Concept**
- **Our model**
- **Experiments and Evaluation**
- **Conclusion**
- **Future Work**

Problem Background

- How to detect users’ locations automatically is significant for many location-based applications such as dietary recommendation and tourism planning.
- There are only about 0.42% posts containing GPS information according to Zhiyuan Cheng et al. You are where you tweet: a content-based approach to geo-locating twitter users 2010, CIKM '10, because most users close their GPS modules. Thus, it’s not easy to detect a user's locations automatically.

Related Work

- Einat Amitay et al. use place name dictionary to identify the main places of a web page in Web-a-where: geotagging web content, SIGIR '04.

↓

- Zhiyuan Cheng et al. find word geographical spatial distribution based on the method of probability in You are where you tweet: a content-based approach to geo-locating twitter users, CIKM '10.

Concept

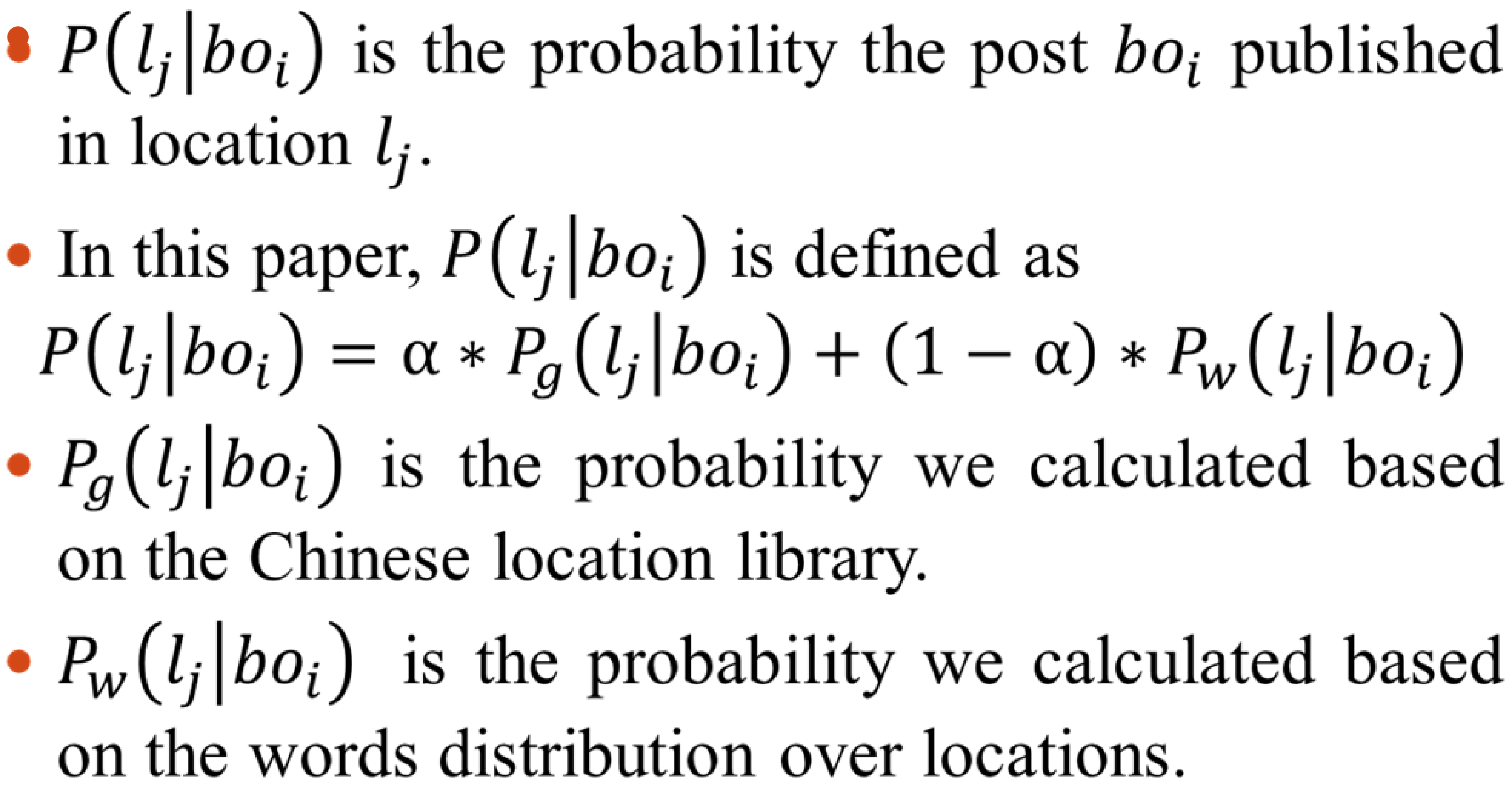
- Location Estimation
  - Systems are designed to detect users’ locations automatically based on users’ behaviors on social media platform.

- Algorithms
  - Content-based method
  - Social-relationship-based method

Our model

- Considering the location names mentioned in the posts can estimate where the user is, but some information (not location names) mentioned in the posts can also reveal where the user is. So, in this paper, we consider location names and other information together to estimate where the user is.

Our model



- $P(l_j|bo_i)$ is the probability the post bo_i published in location l_j.
- In this paper, $P(l_j|bo_i)$ is defined as

$$P(l_j|bo_i) = \alpha * P_g(l_j|bo_i) + (1 - \alpha) * P_w(l_j|bo_i)$$

- $P_g(l_j|bo_i)$ is the probability we calculated based on the Chinese location library.
- $P_w(l_j|bo_i)$ is the probability we calculated based on the words distribution over locations.

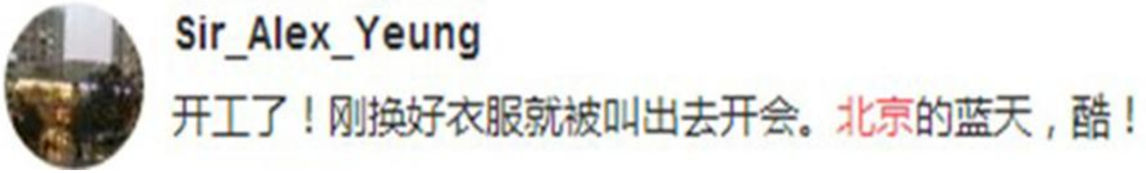
Our model

α is a parameter.

α	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**
F1	0.4685	0.4805	0.4734	0.4713	0.5863	0.5086	0.4688	0.4753	0.4780

- From the table, we can see that when α=0.5, F1 has the best result. So we set α=0.5 in this paper.

Location Library Estimation



Sir_Alex_Yeung

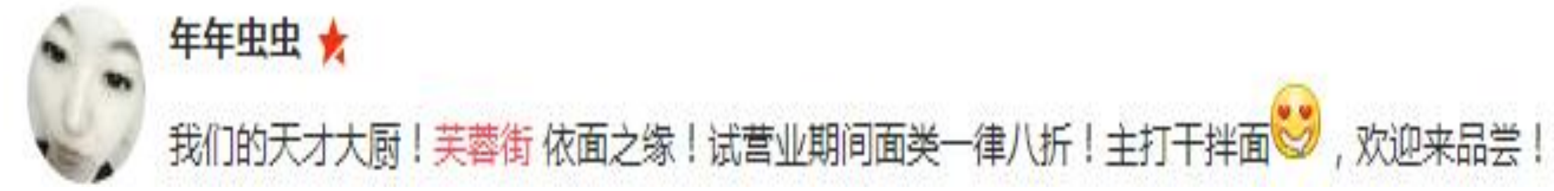
开工了！刚换好衣服就被叫出去开会。北京的蓝天，酷！

- The $P_g(l_j|bo_i)$ is computed as

$$P_g(l_j|bo_i) = \frac{f(l_j)}{\sum_{l_q \in L} f(l_q)}$$

- Where $f(l_j)$ represents the frequency of location l_j mentioned in the post bo_i, $\sum_{l_q \in L} f(l_q)$ is the frequency of all locations mentioned in the post bo_i.

Probability model

- 

  年年虫虫

  我们的天才大厨！芙蓉街 依面之缘！试营业期间面类一律八折！主打干拌面，欢迎来品尝！

- We get all Chinese cities' information from Wiki and use the information to estimate a post published in one location.

$$P_w(l_j|bo_i) = \frac{P(bo_i|l_j) * P(l_j)}{P(bo_i)} \propto P(bo_i|l_j) * P(l_j)$$

Probability model

- $P(bo_i|l_j)$ is calculated in the following equation.

$$P(bo_i|l_j) = \prod_{w_s \in BO_i} P(w_s|l_j)$$

where bo_i is segmented into word set BO_i and w_s is a word of the post bo_i .

Probability model

- The next work is to calculate $P\left(w_s \middle| l_j\right)$. In this paper,

$$P\left(w_s \middle| l_j\right) = \frac{\log count(w_s)}{\sum \log count(w_h)} ; w_{s,} w_h \in W \cap BO_i$$

W is the words set we get from Wiki. $P(l_j)$ is the popularity of location l_j. In this paper,

$$P(l_j) = \frac{\log count(l_j)}{\sum_{l_q \in L} \log count(l_q)}$$

Smooth

- For a user's detected location sequence, we compute the minimum transfer time between two adjacent locations and use it to smooth the sequence in context.
- The threshold time of a person from one location to another:

$$\mathrm{th}(l_j, l_k) = \frac{Dis(l_j, l_k)}{v}$$

Data Set

- Data Set from Sina Weibo

item	number
users	772
posts	826,018
locations	372
posts with locations	304,384

Experimental Results

Methods	Precision	Recall	F1
GL_{lib}	0.8226	0.2029	0.2989
GL_{pm}	0.8394	0.2649	0.3802
GL_{ipm}	0.8335	0.3681	0.4795
Our method	0.8213	0.4991	0.5863

Experimental Results

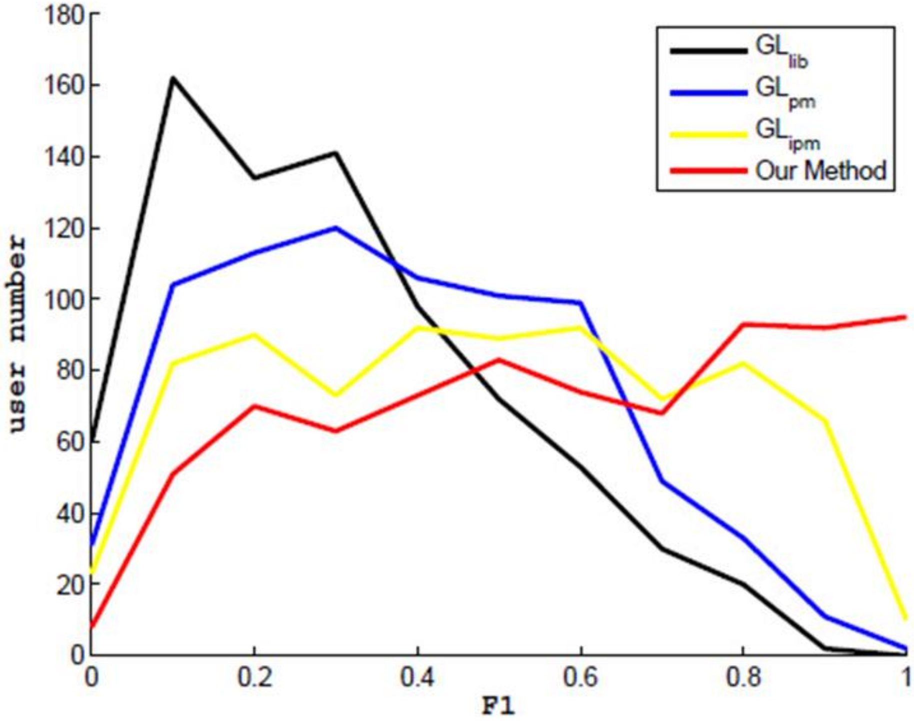


Fig. 1. Distribution of the number of users over $F1$

Conclusion

- Our method considers both the direct matching with location name and the indirect mining of implied word distribution over locations
- The transfer speed between locations is also utilized to smooth the detected location series.
- Experiment verify that our method can outperform the baselines especially in terms of the measure of Recall.

Future work

- Consider social relationship and content together to further improve the performance.
- Detect a movement trajectory for a user.
- Find typical movement patterns and hot spots.
- Predict where the user will go in the next time.

THANKS