Extracting User's Hidden Profile on Twitter

Dong Wang, Mohan Yang, Yuchen Liu

Department of Computer Science
University of California, Los Angeles

Nov 21, 2011

Extracting User's Hidden Profile on Twitter

Introduction

Tiwtter 101

- A world of 140 characters
- Following, follower & non-reciprocal relationship
- A short biography of 160 characters, plaintext
  - Difficult to know a user's profile, including affiliation, occupation, interests etc.
  - 27.2% of users have a bio less than 5 characters, 43.9% of users have a bio less than 10 characters

Introduction

Tiwtter 101

- A world of 140 characters
- Following, follower & non-reciprocal relationship
- A short biography of 160 characters, plaintext
    - Difficult to know a user's profile, including affiliation, occupation, interests etc.
    - 27.2% of users have a bio less than 5 characters, 43.9% of users have a bio less than 10 characters

Benefit of complete user profile

- Recommendation system - related users, news and services
- Advertisement delivery
- Search result relevance

Introduction

Observations

A user is studying at UCLA, but he might not explicitly write this fact in his biography.

- In his followings and followers, there might be a considerate amount of users who is explicitly indicating they are students at UCLA.
- In his tweets, he might post about something related to UCLA. He could also retweet tweets containing such information.

Introduction

Observations

A user is studying at UCLA, but he might not explicitly write this fact in his biography.

- In his followings and followers, there might be a considerate amount of users who is explicitly indicating they are students at UCLA.
- In his tweets, he might post about something related to UCLA. He could also retweet tweets containing such information.

Try to predict whether a user belongs to a category (e.g., studying at UCLA) using the information from

- Followings & followers
- Tweets, location & biography

Extracting User's Hidden Profile on Twitter

Problem Formulation

- A directed graph $G = (V, E)$
- A node $u \in V = \{1, 2, \cdots, n\}$ represents a user in twitter
- A directed edge $(u, v) \in E$ indicates user u is following user v
- Sets $follower(u)$ and $following(u)$
- Size of set is $|follower(u)|$, and $|following(u)|$

Problem Formulation

- A directed graph $G = (V, E)$
- A node $u \in V = \{1, 2, \cdots, n\}$ represents a user in twitter
- A directed edge $(u, v) \in E$ indicates user u is following user v
- Sets $follower(u)$ and $following(u)$
- Size of set is $|follower(u)|$, and $|following(u)|$

- A category $\mathcal{C}$, we want to identify all users that belong to $\mathcal{C}$
- Prior knowledge, $V = \mathcal{A} + \mathcal{B}$
  - Users in $\mathcal{A}$ belong to $\mathcal{C}$
  - The results for users in $\mathcal{B}$ are unknown
- A relevance score s_u for $u \in \mathcal{B}$, rank users in $\mathcal{B}$ based on s_u

Extracting User's Hidden Profile on Twitter

Snowball Algorithm

- The probability of a user u belonging to $\mathcal{C}$ is determined by the relevance score s_u
- Assume the probability of u publishing a tweet belonging to $\mathcal{C}$ is also s_u, and each user publishes the same number (k) of tweets
- The probability of receiving a tweet in $\mathcal{C}$ by u is

$$\frac{\sum_{v \in following(u)} s_v k}{|following(u)| k} = \sum_{v \in following(u)} \frac{s_v}{|following(u)|}$$

- Further assume that a user publishes exactly what he receives, then the probability of publishing a tweet in $\mathcal{C}$ by u is

$$s_u = \sum_{v \in following(u)} \frac{s_v}{|following(u)|}$$

Bidirectional Snowball Algorithm

- Snowball - tweets propagation from user to his followers
- Inverse direction - tweets propagation from user to his followings
- A user in $\mathcal{C}$ tends to follow many users in $\mathcal{C}$, while a user followed by many users in $\mathcal{C}$ tends to belong to $\mathcal{C}$

$$
\begin{aligned}
p_u &= \sum_{v \in following(u)} \frac{p_v}{|following(u)|} \\
q_u &= \sum_{v \in follower(u)} \frac{q_v}{|follower(u)|}
\end{aligned}
$$

- Users are ranked according to the relevance score $s_u = p_u q_u$, which is a combination of relevance to category $\mathcal{C}$ from both following and follower directions

Naive Bayes Algorithm

- Users in $\mathcal{A}$ ($\mathcal{B}$) are positive (negative) training examples
- T_u is the collection of u's tweets, location and biography
- $W = \{w_1, \cdots, w_m\}$ is the word set for corpus $\bigcup_{u=1}^{n} T_u$
- $\mathbf{1}_{T_u}(w_i)$ is the indicator function of T_u

Naive Bayes Algorithm

- Users in $\mathcal{A}$ ($\mathcal{B}$) are positive (negative) training examples
- T_u is the collection of u's tweets, location and biography
- $W = \{w_1, \cdots, w_m\}$ is the word set for corpus $\bigcup_{u=1}^{n} T_u$
- $\mathbf{1}_{T_u}(w_i)$ is the indicator function of T_u
- The naive Bayes classifier finds $i \in \{0, 1\}$ which maximizes

$$p(c = i | w_1 = \mathbf{1}_{T_u}(w_1), \cdots, w_m = \mathbf{1}_{T_u}(w_m)),$$

- or equivalently maximizes

$$p(c = i) \prod_{j=1}^{m} p(w_j = \mathbf{1}_{T_u}(w_j) | c = i).$$

- It is equivalent to determining the sign for s_u

$$\begin{aligned} s_u &= \log(p(c = 1)) - \log(p(c = 0)) \\ &+ \sum_{j=1}^{m} (p(w_j = \mathbf{1}_{T_u}(w_j) | c = 1) - p(w_j = \mathbf{1}_{T_u}(w_j) | c = 0)). \end{aligned}$$

Co-training Algorithm

Perspective of previous algorithms

- Bidirectional snowball algorithm - network structure level
- Naive Bayes algorithm - tweets information level

Co-training Algorithm

Perspective of previous algorithms

- Bidirectional snowball algorithm - network structure level
- Naive Bayes algorithm - tweets information level

Co-training algorithm combines the two algorithms together, iteratively reinforce the result of one algorithm by the result of the other algorithm.

Co-training Algorithm

Algorithm 1 CO-TRAINING

Input: Category $\mathcal{C}$, two disjoint sets $\mathcal{A}$ and $\mathcal{B}$, parameter k and l
Output: An array *rank* containing users in $\mathcal{B}$ ranked on the probability of belonging to $\mathcal{C}$

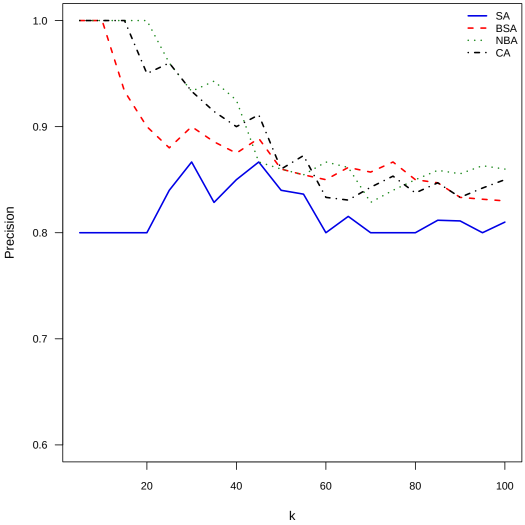
```
repeat
    rank' ← bidirectional snowball algorithm(𝒜, ℬ)
    rank ← naive Bayes algorithm(𝒜, ℬ)
    𝒜 ← 𝒜 + {top l users in rank'}
    𝒜 ← 𝒜 + {top l users in rank}
until Top k users in rank' and rank are the same
return rank
```

Extracting User's Hidden Profile on Twitter

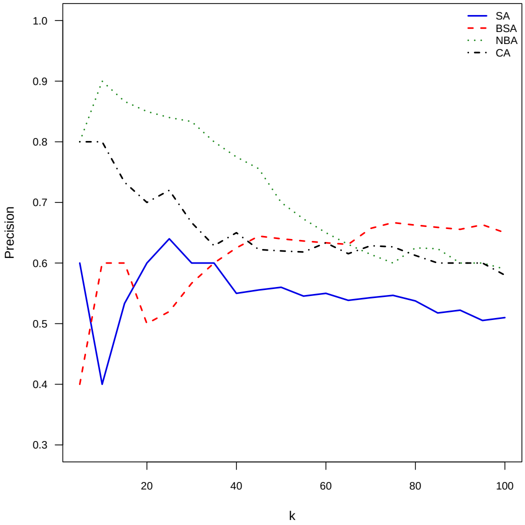
Experiment Setup

- Data collection
  - Seed users from UCLA, USC, Stanford and MIT, two-level breadth first traversal starting from seed user
  - $540,000$ users, $15,321,508$ tweets, $3,143,115$ different words
  - Filter out less frequent words(occurrance < 100) $=>$ about $20,000$ words
- Category $\mathcal{C}$ and keyword $z_{\mathcal{C}} => V = \mathcal{A} + \mathcal{B}$
  - $\mathcal{C} =$ "users in UCLA", $z_{\mathcal{C}} =$ "UCLA"
- Randomly select 20% from $\mathcal{A}$, remove the bigoraphy and move them to $\mathcal{B}$
- Manually label top 100 results of different methods for UCLA category

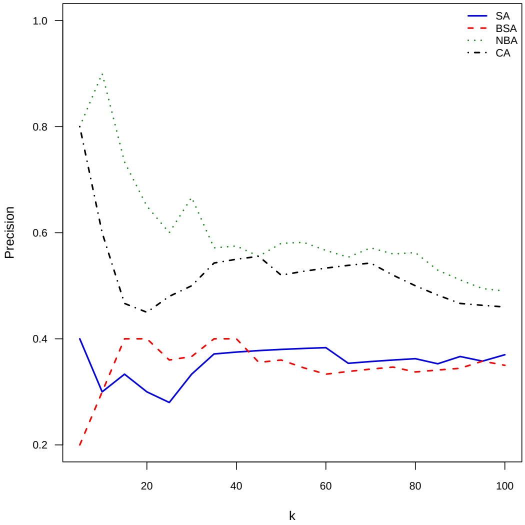
Precision@k for UCLA Category



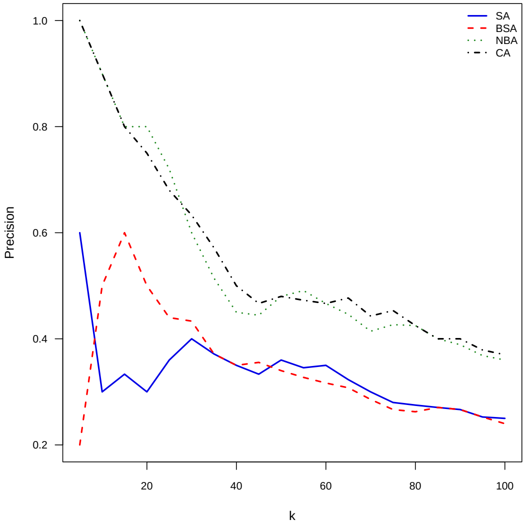
Precision@k for USC Category



Precision@k for Stanford Category

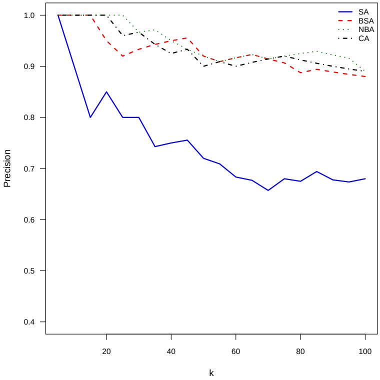


Precision@*k* for MIT Category

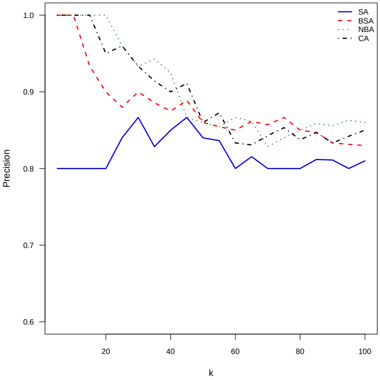


3rd Ranking User from Snowball Algorithm for UCLA Category

Human Labeled Data Vs. Automatic Evaluation

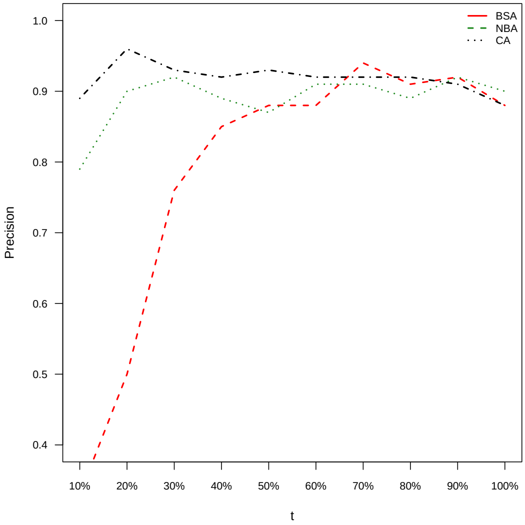


Precision@k for UCLA category on human labeled data



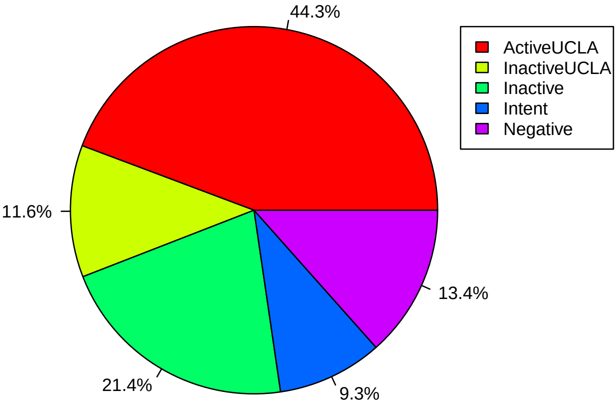
Precision@k for UCLA category with automatic evaluation

Precision@*k* for UCLA Category With Loss of Information

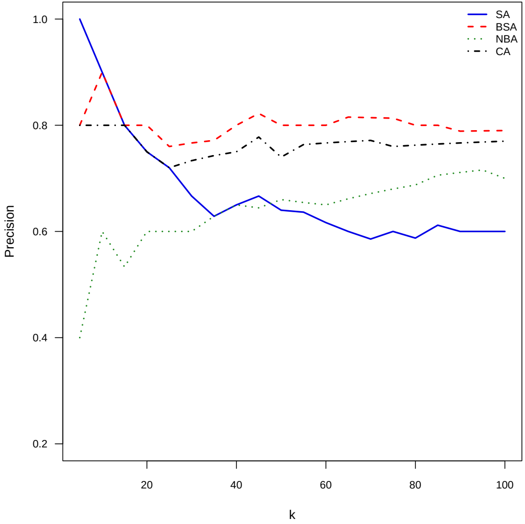


Extracting User's Hidden Profile on Twitter

User Classification



Precision@k of Active Users in UCLA Category



Extracting User's Hidden Profile on Twitter

Conclusion

- Three simple algorithms and a co-training algorithm to rank the users based on the relevance score to a given category
- These simple algorithms perform very well on twitter data
- Co-training algorithm can be applied to many other problems that require learning to rank the nodes in a graph

Future work

- Different weights for users based on the importance and activity in the network
- Apply co-training algorithm to friend recommendation system

Thanks!