Inferring User Political Preferences from Streaming Communications

Svitlana Volkova,[1] **Glen Coppersmith**[2] **and Benjamin Van Durme**[1,2]
[1]Center for Language and Speech Processing,
[2]Human Language Technology Center of Excellence,
Johns Hopkins University, Baltimore, MD 21218
svitlana@jhu.edu, coppersmith@jhu.edu, vandurme@cs.jhu.edu

Abstract

Existing models for social media personal analytics assume access to thousands of messages per user, even though most users author content only sporadically over time. Given this sparsity, we: (i) leverage content from the local neighborhood of a user; (ii) evaluate batch models as a function of size and the amount of messages in various types of neighborhoods; and (iii) estimate the amount of time and tweets required for a dynamic model to predict user preferences. We show that even when limited or no self-authored data is available, language from friend, retweet and user mention communications provide sufficient evidence for prediction. When updating models over time based on Twitter, we find that political preference can be often be predicted using roughly 100 tweets, depending on the context of user selection, where this could mean hours, or weeks, based on the author's tweeting frequency.

1 Introduction

Inferring latent user attributes such as gender, age, and political preferences (Rao et al., 2011; Zamal et al., 2012; Cohen and Ruths, 2013) automatically from personal communications and social media including emails, blog posts or public discussions has become increasingly popular with the web getting more social and volume of data available. Resources like Twitter[1] or Facebook[2] become extremely valuable for studying the underlying properties of such informal communications because of its volume, dynamic nature, and diverse population (Lunden, 2012; Smith, 2013).

The existing batch models for predicting latent user attributes rely on thousands of tweets per author (Rao et al., 2010; Conover et al., 2011; Pennacchiotti and Popescu, 2011a; Burger et al., 2011; Zamal et al., 2012; Nguyen et al., 2013). However, most Twitter users are less prolific than those examined in these works, and thus do not produce the thousands of tweets required to obtain their levels of accuracy e.g., the median number of tweets produced by a random Twitter user per day is 10. Moreover, recent changes to Twitter API querying rates further restrict the speed of access to this resource, effectively reducing the amount of data that can be collected in a given time period.

In this paper we analyze and go beyond static models formulating personal analytics in social media as a streaming task. We first evaluate batch models that are cognizant of low-resource prediction setting described above, maximizing the efficiency of content in calculating personal analytics. To the best of our knowledge, this is the first work that makes explicit the tradeoff between accuracy and cost (manifest as calls to the Twitter API), and optimizes to a different tradeoff than state-of-the-art approaches, seeking maximal performance when limited data is available. In addition, we propose streaming models for personal analytics that dynamically update user labels based on their stream of communications which has been addressed previously by Van Durme (2012b). Such models better capture the real-time nature of evidence being used in latent author attribute predictions tasks. Our main contributions include:

- develop low-resource and real-time dynamic approaches for personal analytics using as an example the prediction of political preference of Twitter users;
- examine the relative utility of six different notions of "similarity" between users in an implicit Twitter social network for personal analytics;

[1] http://www.demographicspro.com/
[2] http://www.wolframalpha.com/facebook/

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, pages 186–196,
Baltimore, Maryland, USA, June 23-25 2014.

- experiments are performed across multiple datasets supporting the prediction of political preference in Twitter, to highlight the significant differences in performance that arise from the underlying collection and annotation strategies.

2 Identifying Twitter Social Graph

Twitter users interact with one another and engage in direct communication in different ways e.g., using retweets, user mentions e.g., *@youtube* or hashtags e.g., *#tcot*, in addition to having explicit connections among themselves such as following, friending. To investigate all types of social relationships between Twitter users and construct Twitter social graphs we collect lists of followers and friends, and extract user mentions, hashtags, replies and retweets from communications.[3]

2.1 Social Graph Definition

Lets define an attributed, undirected graph $G = (V, E)$, where V is a set of vertices and E is a set of edges. Each vertex v_i represents someone in a communication graph i.e., *communicant*: here a Twitter user. Each vertex is attributed with a feature vector $\vec{f}(v_i)$ which encodes communications e.g., tweets available for a given user. Each vertex is associated with a latent attribute $a(v_i)$, in our case it is binary $a(v_i) \in \{D, R\}$, where D stands for Democratic and R for Republican users. Each edge $e_{ij} \in E$ represents a connection between v_i and v_j, $e_{ij} = (v_i, v_j)$ and defines different social circles between Twitter users e.g., follower (f), friend (b), user mention (m), hashtag (h), reply (y) and retweet (w). Thus, $E \in V^{(2)} \times \{f, b, h, m, w, y\}$. We denote a set of edges of a given type as $\phi_r(E)$ for $r \in \{f, b, h, m, w, y\}$. We denote a set of vertices adjacent to v_i by social circle type r as $N_r(v_i)$ which is equivalent to $\{v_j \mid e_{ij} \in \phi_r(E)\}$. Following Filippova (2012) we refer to $N_r(v_i)$ as v_i's social circle, otherwise known as a neighborhood. In most cases, we only work with a sample of a social circle, denoted by $N'_r(v_i)$ where $|N'_r(v_i)| = k$ is its size for v_i.

Figure 1 presents an example of a social graph derived from Twitter. Notably, users from different social circles can be shared across the users of the same or different classes e.g., a user v_j can be in both follower circle $v_j \in N_f(v_i), v_i \in D$ and retweet circle $v_j \in N_w(v_k), v_k \in R$.

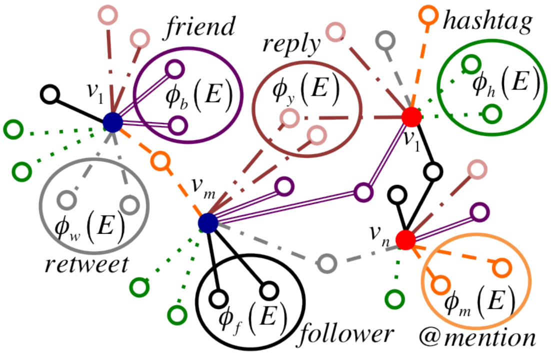


Figure 1: An example of a social graph with follower, friend, @mention, reply, retweet and hashtag social circles for each user of interest e.g., blue: Democratic, red: Republican.

2.2 Candidate-Centric Graph

We construct candidate-centric graph G_{cand} by looking into following relationships between the users and Democratic or Republican candidates during the 2012 US Presidential election. In the Fall of 2012, leading up to the elections, we randomly sampled $n = 516$ Democratic and $m = 515$ Republican users. We labeled users as Democratic if they exclusively follow both Democratic candidates[4] – BarackObama and JoeBiden but do not follow both Republican candidates – MittRomney and RepPaulRyan and vice versa. We collectively refer to D and R as our "users of interest" for which we aim to predict political preference. For each such user we collect recent tweets and randomly sample their immediate $k = 10$ neighbors from follower, friend, user mention, reply, retweet and hashtag social circles.

2.3 Geo-Centric Graph

We construct a geo-centric graph G_{geo} by collecting $n = 135$ Democratic and $m = 135$ Republican users from the Maryland, Virginia and Delaware region of the US with self-reported political preference in their biographies. Similar to the candidate-centric graph, for each user we collect recent tweets and randomly sample user social circles in the Fall of 2012. We collect this data to get a sample of politically less active users compared to the users from candidate-centric graph.

2.4 ZLR Graph

We also consider a G_{ZLR} graph constructed from a dataset previously used for political affiliation

[3] The code and detailed explanation on how we collected all six types of user neighbors and their communications using Twitter API can be found here: http://www.cs.jhu.edu/ svitlana/

[4] As of Oct 12, 2012, the number of followers for Obama, Biden, Romney and Ryan were 2m, 168k, 1.3m and 267k.

classification (Zamal et al., 2012). This dataset consists of 200 Republican and 200 Democratic users associated with 925 tweets on average per user.[5] Each user has on average 6155 friends with 642 tweets per friend. Sharing restrictions and rate limits on Twitter data collection only allowed us to recreate a semblance of ZLR data[6] – 193 Democratic and 178 Republican users with 1K tweets per user, and 20 neighbors of four types including follower, friends, user mention and retweet with 200 tweets per neighbor for each user of interest.

3 Batch Models

Baseline User Model As input we are given a set of vertices representing users of interest $v_i \in V$ along with feature vectors $\vec{f}(v_i)$ derived from content authored by the user of interest. Each user is associated with a non-zero number of publicly posted tweets. Our goal is assign to a category each user of interest v_i based on $\vec{f}(v_i)$. Here we focus on a binary assignment into the categories Democratic D or Republican R. The log-linear model[7] for such binary classification is:

$$\Phi_{v_i} = \begin{cases} D & (1+\exp[-\vec{\theta}\cdot\vec{f}(v_i)])^{-1} \geq 0.5, \\ R & \text{otherwise.} \end{cases} \tag{1}$$

where features are normalized word ngram counts extracted from v_i's tweets $\vec{f_t}(v_i) : D \times t(v_i) \rightarrow \mathbb{R}$.

The proposed baseline model follows the same trends as the existing state-of-the-art approaches for user attribute classification in social media as described in Section 8. Next we propose to extend the baseline model by taking advantage of language in user social circles as describe below.

Neighbor Model As input we are given user-local neighborhood $N_r(v_i)$, where r is a neighborhood type. Besides the neighborhood's type r, each is characterized by:

- the number of communications per neighbor $\vec{f_t}(N_r)$, $t = \{5, 10, 15, 25, 50, 100, 200\}$;
- the order of the social circle – the number of neighbors per user of interest $|N_r| = deg(v_i)$, $n = \{1, 2, 5, 10\}$.

Our goal is to classify users of interest using evidence (e.g., communications) from their local neighborhood $\sum\limits_n \vec{f_t}[N_r(v_i)] \equiv \vec{f}(N_r)$ as Democratic or Republican. The corresponding log-linear model is defined as:

$$\Phi_{N_r} = \begin{cases} D & (1+\exp[-\vec{\theta}\cdot\vec{f}(N_r)])^{-1} \geq 0.5, \\ R & \text{otherwise.} \end{cases} \tag{2}$$

To check whether our static models are cognizant of low-resource prediction settings we compare the performance of the user model from Eq.1 and the neighborhood model from Eq.2. Following the streaming nature of social media, we see the scarce available resource as the number of requests allowed per day to the Twitter API. Here we abstract this to a model assumption where we receive one tweet t_k at a time and aim to maximize classification performance with as few tweets per user as possible:[8]

- for the baseline user model:

$$\underset{k}{\text{minimize}} \quad \sum_k t_k(v_i), \tag{3}$$

- for the neighborhood model:

$$\underset{k}{\text{minimize}} \quad \sum_n \sum_k t_k[N_r(v_i)]. \tag{4}$$

4 Streaming Models

We rely on straightforward Bayesian rule update to our batch models in order to simulate a real-time streaming prediction scenario as a first step beyond the existing models as shown in Figure 2.

The model makes predictions of a latent user attribute e.g., Republican under a model assumption of sequentially arriving, independent and identically distributed observations $T = (t_1, \ldots, t_k)^9$. The model dynamically updates posterior probability estimates $p(a(v_i) = R|t_k)$ for a given user

[5] The original dataset was collected in 2012 and has been recently released at http://icwsm.cs.mcgill.ca/. Political labels are extracted from http://www.wefollow.com as described by Pennacchiotti and Popescu (2011b).

[6] This inability to perfectly replicate prior work based on Twitter is a recognized problem throughout the community of computational social science, arising from the data policies of Twitter itself, it is not specific to this work.

[7] We use log-linear models over reasonable alternatives such as perceptron or SVM, following the practice of a wide range of previous work in related areas (Smith, 2004; Liu et al., 2005; Poon et al., 2009) including text classification in social media (Van Durme, 2012b; Yang and Eisenstein, 2013).

[8] The separate issue is that many authors simply don't tweet very often. For instance, 85.3% of all Twitter users post less than one update per day as reported at http://www.sysomos.com/insidetwitter/. Thus, their communications are scare even if we could get all of them without rate limiting from Twitter API.

[9] Given the dynamic character of online discourse it will clearly be of interest in the future to consider models that go beyond the iid assumption.

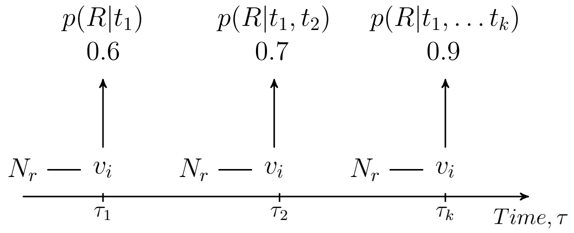


Figure 2: Stream-based classification of an attribute $a(v_i) \in \{R, D\}$ given a stream of communications $t_1, t_2, \dots, t_k$ authored by a user v_i or user immediate neighbors from N_r social circles at time $\tau_1, \tau_2, \dots, \tau_k$.

v_i as an additional evidence t_k is acquired, as defined in a general form below for any latent attribute $a(v_i) \in A$ given the tweets T of user v_i:

$$p(a(v_i) = x \in A \mid T) = \frac{p(T \mid a(v_i) = x) \cdot p(a(v_i) = x)}{\sum_{y \in A} p(T \mid a(v_i) = y) \cdot p(a(v_i) = y)} = \frac{\prod_k p(t_k \mid a(v_i) = x) \cdot p(a(v_i) = x)}{\sum_{y \in A} \prod_k p(t_k \mid a(v_i) = y) \cdot p(a(v_i) = y),} \quad (5)$$

where y is the number of all possible attribute values, and k is the number of tweets per user.

For example, to predict user political preference, we start with a prior $P(R) = 0.5$, and sequentially update the posterior $p(R \mid T)$ by accumulating evidence from the likelihood $p(t_k|R)$:

$$p(R \mid T) = \frac{\prod_k p(t_k|R) \cdot p(R)}{\prod_k P(t_k|R) \cdot p(R) + \prod_k P(t_k|D) \cdot p(D).} \quad (6)$$

Our goal is to maximize posterior probability estimates given a stream of communications for each user in the data over (a) time τ and (b) the number of tweets T. For that, for each user we take tweets that arrive continuously over time and apply two different streaming models:

- **User Model with Dynamic Updates:** relies exclusively on user tweets $t_1^{(v_i)}, \dots, t_k^{(v_i)}$ following the order they arrive over time τ, where for each user v_i we dynamically update the posterior $p(R \mid t_1^{(v_i)}, \dots, t_k^{(v_i)})$.
- **User-Neighbor Model with Dynamic Updates:** relies on both neighbor N_r communications including friend, follower, retweet, user mention and user tweets $t_1^{(v_i)}, \dots, t_k^{(N_r)}$ following the order they arrive over time τ; here we dynamically update the posterior probability $p(R \mid t_1^{(v_i)}, \dots, t_k^{(N_r)})$.

5 Experimental Setup

We design a set of experiments to analyze static and dynamic models for political affiliation classification defined in Sections 3 and 4.

5.1 Batch Classification Experiments

We first answer whether communications from user-local neighborhoods can help predict political preference for the user. To explore the contribution of different neighborhood types we learn static user and neighbor models on G_{cand}, G_{geo} and G_{ZLR} graphs. We also examine the ability of our static models to predict user political preferences in low-resource setting e.g., 5 tweets.

The existing models follow a standard setup when either user or neighbor tweets are available during train and test. For a static neighbor model we go beyond that, and train our the model on all data available per user, but only apply part of the data at the test time, pushing the boundaries of how little is truly required for classification. For example, we only use follower tweets for G^{test}, but we use tweets from all types of neighbors for G^{train}. Such setup will simulate different real-world prediction scenarios which have not been previously explored, to our knowledge e.g., when a user has a private profile or has not tweeted yet, and only user neighbor tweets are available.

We experiment with our static neighbor model defined in Eq.2 with the aim to:

1. evaluate neighborhood size influence, we change the number of neighbors and try $n = [1, 2, 5, 10]$ neighbor(s) per user;
2. estimate neighbor content influence, we alternate the amount of content per neighbor and try $t = [5, 10, 15, 25, 50, 100, 200]$ tweets.

We perform 10-fold cross validation[10] and run 100 random restarts for every n and t parameter combination. We compare our static neighbor and user models using the cost functions from Eq.3 and Eq.4. For all experiments we use `LibLinear` (Fan et al., 2008), integrated in the `Jerboa` toolkit (Van Durme, 2012a). Both models defined in Eq.1 and Eq.2 are learned using normalized count-based word ngram features extracted from either user or neighbor tweets.[11]

[10] For each fold we split the data into 3 parts: 70% train, 10% development and 20% test.

[11] For brevity we omit reporting results for bigram and trigram features, since unigrams showed superior performance.

5.2 Streaming Classification Experiments

We evaluate our models with dynamic Bayesian updates on a continuous stream of communications over time as shown in Figure 2. Unlike static model experiments, we are not modeling the influence of the number of neighbors or the amount of content per neighbor. Here, we order user and neighbor communication streams by real world time of posting and measure changes in posterior probabilities over time. The main purpose of these experiments is to quantitatively evaluate (1) the number of tweets and (2) the amount of real world time it takes to observe enough evidence on Twitter to make reliable predictions.

We experiment with log-linear models defined in Eq. 1 and 2 and continuously estimate the posterior probabilities $P(R \mid T)$ as defined in Eq.6. We average the posterior probability results over the users in G_{cand}, G_{geo} and G_{ZLR} graphs. We train streaming models on an attribute balanced subset of tweets for each user v_i excluding v_i's tweets (or v_i's neighbor tweets for a joint model). This setup is similar to leave-one-out classification. The classifier is learned using binary word ngram features extracted from user or user-neighbor communications. We prefer binary to normalized count-based features to overcome sparsity issues caused by making predictions on each tweet individually.

6 Static Classification Results

6.1 Modeling User Content Influence

We investigate classification decision probabilities for our static user model Φ_{v_i} by making predictions on a random set of 5 vs. 100 tweets per user. To our knowledge only limited work on personal analytics (Burger et al., 2011; Van Durme, 2012b) have performed this straight-forward comparison. For that purpose, we take a random partition containing 100 users of G_{cand} graph and perform four independent classification experiments – two runs using 5 and two runs using 100 tweets per user.

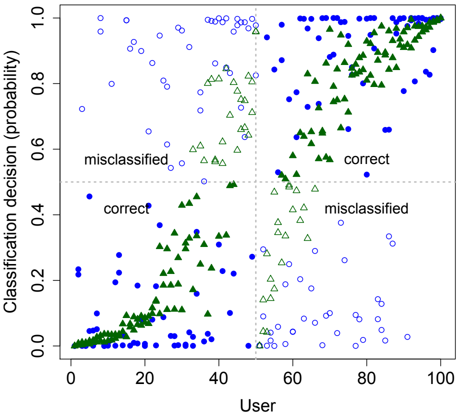
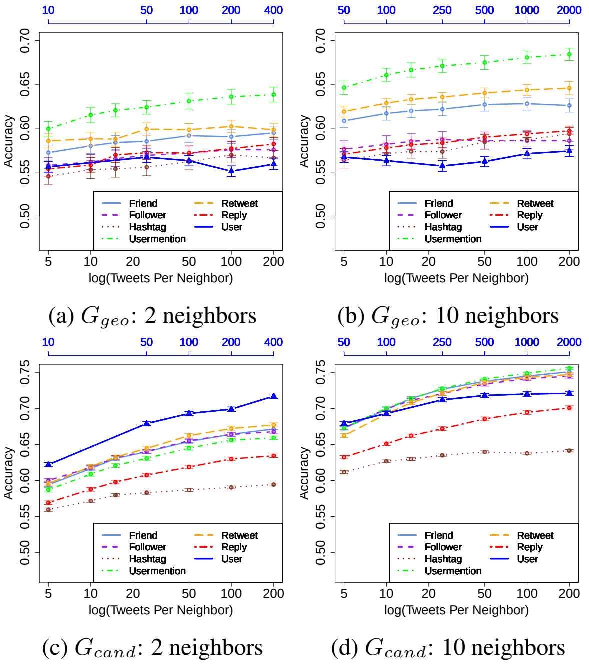


Figure 3: Classification probabilities for Φ_{v_i} estimated over 100 users in G_{cand} tested on 5 (blue) vs. 100 (green) tweets per user where Republican = 1, Democratic = 0, filled markers = correctly classified, not filled = misclassified users.



(a) G_{geo}: 2 neighbors (b) G_{geo}: 10 neighbors

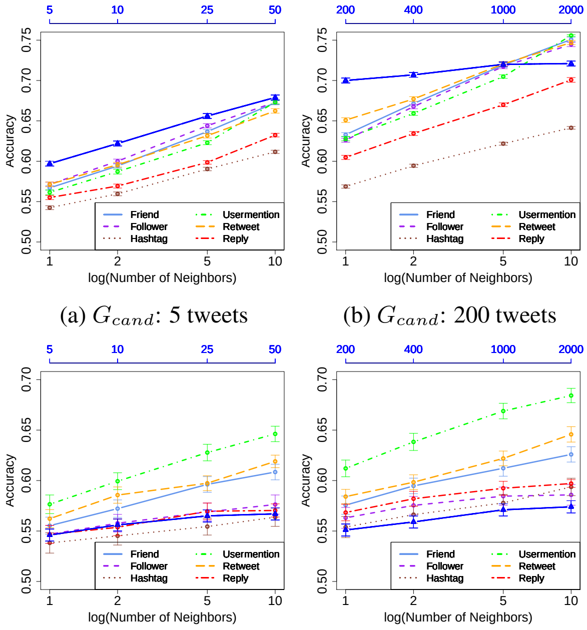
(c) G_{cand}: 2 neighbors (d) G_{cand}: 10 neighbors

Figure 4: Modeling the influence of the number of tweets per neighbor t=[5, .., 200] for G_{cand} and G_{geo} graphs.

Figure 3 demonstrates that more tweets during prediction time lead to higher accuracy by showing that more users with 100 tweets are correctly classified e.g., filled green markers in the right upper quadrant are true Republicans and in the left lower quadrant are true Democrats. Moreover, a lot of users with 100 tweets are close to 0.5 decision probability which suggests that the classifier is just uncertain rather then being completely off, e.g., misclassified Republican users with 5 tweets (not filled blue markers in the right lower quadrant) are close to 0. These results follow naturally from the underlying feature representation: having more tweets per user leads to a lower variance estimate of a target multinomial distribution. The more robustly this distribution is estimated (based on having more tweets) the more confident we should be in the classifier output.

6.2 Modeling Neighbor Content Influence

Here we discuss the results for our static neighborhood model. We study the influence of the neighborhood type r and size in terms of the number of neighbors n and tweets t per neighbor.



(a) G_{cand}: 5 tweets

(b) G_{cand}: 200 tweets

(c) G_{geo}: 5 tweets

(d) G_{geo}: 200 tweets

Figure 5: Modeling the influence of the number of neighbors per user n=$[1, .., 10]$ for G_{cand} and G_{geo} graphs.

In Figure 4 we present accuracy results for G_{cand} and G_{geo} graphs. Following Eq.3 and 4, we spent an equal amount of resources to obtain 100 user tweets and 10 tweets from 10 neighbors. We annotate these 'points of equal number of communications' with a line on top marked with a corresponding number of user tweets.

We show that three of six social circles – friend, retweet and user-mention yield better accuracy compared to the user model for all graphs when $t \geq 250$. Thus, for effectively classifying a given user v_i it is better to take 200 tweets each from 10 neighbors rather than 2,000 tweets from the user.

The best accuracy for G_{cand} is 0.75 for friend, follower, retweet and user-mention neighborhoods which is 0.03 higher than the user baseline; for G_{geo} is 0.67 for user-mention and 0.64 for retweet circles compared to 0.57 for the user model; for G_{ZLR} is 0.863 for retweet and 0.849 for friend circles which is 0.11 higher that the user baseline. Finally, similarly to the results for the user model given in Figure 3, increasing the number of tweets per neighbor from 5 to 200 leads to a significant gain in performance for all neighborhood types.

6.3 Modeling Neighborhood Size

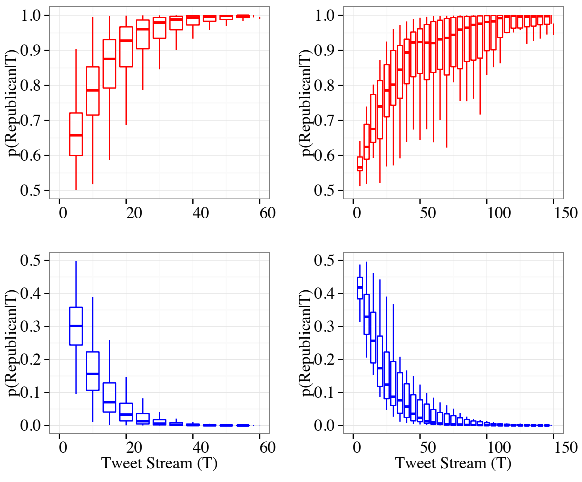
In Figure 5 we present accuracy results to show neighborhood size influence on classification performance for G_{geo} and G_{cand} graphs. Our results demonstrate that even small changes to the neighborhood size n lead to better performance which does not support the claims by Zamal et al. (2012). We demonstrate that increasing the size of the neighborhood leads to better performance across six neighborhood types. Friend, user mention and retweet neighborhoods yield the highest accuracy for all graphs. We observe that when the number of neighbors is $n = 1$, the difference in accuracy across all neighborhood types is less significant but for $n \geq 2$ it becomes more significant.

7 Streaming Classification Results

7.1 Modeling Dynamic Posterior Updates from a User Stream

Figures 6a and 6b demonstrate dynamic user model prediction results averaged over users from G_{cand} and G_{ZLR} graphs. Each figure outlines changes in sequential average probability estimates $p_{\mu}(R \mid T)$ for each individual self-authored tweet t_k as defined in Eq. 6. The average probability estimates $p_{\mu}(R \mid T)$ are reported for every 5 tweets in a stream $T = (t_1, \ldots t_k)$ as $\frac{\sum_n P(R|t_k)}{n}$, where n is the total number of users with the same attribute R or D. We represent $p_{\mu}(R \mid T)$ as a box and whisker plot with the median, lower and upper quantiles to show the variance; the length of whiskers indicate lower and upper extreme values.

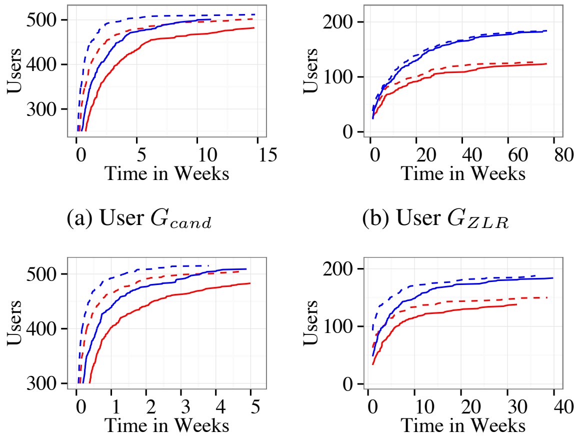
We find similar behavior across all three graphs. In particular, the posterior estimates converge faster when predicting Democratic than Republican users but it has been trained on an equal number of tweets per class. We observe that average posterior estimates $P_{\mu}(R \mid T)$ converge faster to 0



(a) User G_{cand}

(b) User G_{ZLR}

Figure 6: Streaming classification results from user communications for G_{cand} and G_{ZLR} graphs averaged over every 5 tweets (red - Republican, blue - Democratic).



(a) User G_{cand}

(b) User G_{ZLR}

(c) User-Neigh G_{cand}

(d) User-Neigh G_{ZLR}

Figure 7: Time needed for (a) - (b) dynamic user model and (c) - (d) joint user-neighbor model to infer political preferences of Democratic (blue) and Republican (red) users at 75% (dotted line) and 95% (solid line) accuracy levels.

(Democratic) than to 1 (Republican) in Figures 6a and 6b. It suggests that language of Democrats is more expressive of their political preference than language of Republicans. For example, frequent politically influenced terms used widely by Democratic users include *faith4liberty, constitutionally, pass, vote2012, terroristic.*

The variance for average posterior estimates decreases when the number of tweets increases for all three datasets. Moreover, we detect that $P_\mu(R|T)$ estimates for users in G_{cand} converge 2-3 times faster in terms of number of tweets than for users in G_{ZLR}. The lowest convergence is detected for G_{geo} where after $t_k = 250$ tweets the average posterior estimate $P_\mu(R \mid t_k) = 0.904 \pm 0.044$ and $P_\mu(D \mid t_k) = 0.861 \pm 0.008$. It means that users in G_{cand} are more politically vocal compared to users in G_{ZLR} and G_{geo}. As a result, less active users in G_{geo} just need more than 250 tweets to converge to a true 0 or 1 class. These results are coherent with the outcomes for our static models shown in Figures 4 and 5. These findings further confirm that differences in performance are caused by various biases present in the data due to distinct sampling and annotation approaches.

Figure 7a and 7b illustrate the amount of time required for the user model to infer political preferences estimated for 1,031 users in G_{cand} and 371 users in G_{ZLR}. The amount of time needed can be evaluated for different accuracy levels e.g., 0.75 and 0.95. Thus, with 75% accuracy we classify:

- 100 ($\sim$20%) Republican users in 3.6 hours and Democratic users in 2.2 hours for G_{cand};
- 100 ($\sim$56%) R users in 20 weeks and 100 ($\sim$52%) D users in 8.9 weeks for G_{ZLR} which is 800 times longer that for G_{cand};
- 100 ($\sim$75%) R users in 12 weeks and 80 ($\sim$60%) D users in 19 weeks for G_{geo}.

Such extreme divergences in the amount of time required for classification across all graphs should be of strong interest to researchers concerned with latent attribute prediction tasks because Twitter users produce messages with extremely different frequencies. In our case, users in G_{ZLR} tweet approximately 800 times less frequently than users in G_{cand}.

7.2 Modeling Dynamic Posterior Updates from a Joint User-Neighbor Stream

We estimate dynamic posterior updates from a joint stream of user and neighbor communications in G_{geo}, G_{cand} and G_{ZLR} graphs. To make a fair comparison with a streaming user model, we start with the same user tweet $t_0(v_i)$. Then instead of waiting for the next user tweet we rely on any neighbor tweets that appear until the user produces the next tweet $t_1(v_i)$. We rely on communications from four types of neighbors such as friends, followers, retweets and user mentions.

The convergence rate for the average posterior probability estimates $P_\mu(R|T)$ depending on the number of tweets is similar to the user model results presented in Figure 6. However, for G_{geo} the variance for $P_\mu(R|T)$ is higher for Democratic users; for G_{ZLR} $P_\mu(R|T) \rightarrow 1$ for Republicans in less than 110 tweets which is $\Delta t = 40$ tweets faster than the user model; for G_{cand} the convergence for both $P_\mu(R|T) \rightarrow 1$ and $P_\mu(D|T) \rightarrow 0$ is not significantly different than the user model.

Figures 7c and 7d show the amount of time required for a joint user-neighbor model to infer political preferences estimated for users in G_{cand} and G_{ZLR}. We find that with 75% accuracy we can classify 100 users for:

- G_{cand}: Republican users in 23 minutes and Democratic users in 10 minutes;
- G_{ZLR}: R users in 3.2 weeks and D users in 1.1 weeks which is 7 times faster on average across attributes than for the user model;
- G_{geo}: R users in 1.2 weeks and D users in 3.5 weeks which is on average 6 times faster across attributes than for the user model.

Similar or better $P_\mu(R|T)$ convergence in terms of the number of tweets and, especially, in the amount of time needed for user and user-neighbor models further confirms that neighborhood content is useful for political preference prediction. Moreover, communications from a joint stream allow to make an inference up to 7 times faster.

8 Related Work

Supervised Batch Approaches The vast majority of work on predicting latent user attributes in social media apply supervised static SVM models for discrete categorical e.g., gender and regression models for continuous attributes e.g., age with lexical bag-of-word features for classifying user gender (Garera and Yarowsky, 2009; Rao et al., 2010; Burger et al., 2011; Van Durme, 2012b), age (Rao et al., 2010; Nguyen et al., 2011; Nguyen et al., 2013) or political orientation. We present an overview of the existing models for political preference prediction in Table 1.

Bergsma et al. (2012) following up on Rao's work (2010) on adding socio-linguistic features to improve gender, ethnicity and political preference prediction show that incorporating stylistic and syntactic information to the bag-of-word features improves gender classification.

Other methods characterize Twitter users by applying limited amounts of network structure information in addition to lexical features. Conover et al. (2011) rely on identifying strong partisan clusters of Democratic and Republican users in a Twitter network based on retweet and user mention degree of connectivity, and then combine this clustering information with the follower and friend neighborhood size features. Pennacchiotti et al. (2011a; 2011b) focus on user behavior, network structure and linguistic features. Similar to our work, they assume that users from a particular class tend to reply and retweet messages of the users from the same class. We extend this assumption and study other relationship types e.g., friends, user mentions etc. Recent work by Wong et al. (2013) investigates tweeting and retweeting behavior for political learning during 2012 US Presidential election. The most similar work to ours is by Zamal et al. (2012), where the authors apply features from the tweets authored by a user's friend to infer attributes of that user. In this paper, we study different types of user social circles in addition to a friend network.

Approach	Users	Tweets	Features	Accur.
Rao et al. (2010)	1K	2M	ngrams	**0.824**
			socio-ling	0.634
			stacked	0.809
Pennacchiotti and Popescu (2011a)	10.3K	–	ling-all	0.770
			soc-all	0.863
			full	**0.889**
Conover et al. (2011)	1,000	1M	full-text	0.792
			hashtags	0.908
			clusters	**0.949**
Zamal et al. (2012)	400	400K	UserOnly	0.890
		3.85M	Nbr	0.920
		4.25M	User-Nbr[11]	**0.932**
Cohen and Ruths (2013)	397	397K	features from (Zamal et al., 2012)	**0.910**
	1.8K	1.8M		0.840
	262	262K		0.680
	196	196K		0.870
This paper (batch classification)	G_{cand}	206K	user ngrams	0.720
	1,031	2M	neighbor	**0.750**
	G_{geo}	54K	user ngrams	0.570
	270	540K	neighbor	**0.670**
	G_{ZLR}	371K	user ngrams	0.886
	371	1.5M	neighbor	**0.920**
This paper (dynamic Bayesian update classification)	G_{cand}	103K	user stream	0.995
	1,031	130K	user-neigh.	**0.999**
	G_{geo}	54K	user stream	0.843
	270	67K	user-neigh.	**0.882**
	G_{ZLR}	74K	user stream	0.892
	371	185K	user-neigh.	**0.999**

Table 1: Overview of the existing approaches for political preference classification in Twitter.

Additionally, using social media for mining political opinions (O'Connor et al., 2010a; Maynard and Funk, 2012) or understanding sociopolitical trends and voting outcomes (Tumasjan et al., 2010; Gayo-Avello, 2012; Lampos et al., 2013) is becoming a common practice. For instance, Lampos et al. (2013) propose a bilinear user-centric model for predicting voting intentions in the UK and Australia from social media data. Other works explore political blogs to predict what content will get the most comments (Yano et al., 2013) or analyze communications from Capitol Hill[12] to predict campaign contributors based on this content (Yano and Smith, 2013).

Unsupervised Batch Approaches Bergsma et al. (2013) show that large-scale clustering of user names improves gender, ethnicity and location classification on Twitter. O'Connor et al. (2010b) following the work by Eisenstein (2010) propose a Bayesian generative model to discover demographic language variations in Twitter. Rao et al. (2011) suggest a hierarchical Bayesian model which takes advantage of user name morphology for predicting user gender and ethnicity. Golbeck et al. (2010) incorporate Twitter data in a spatial model of political ideology.

Streaming Approaches Van Durme (2012b) proposed streaming models to predict user gender in Twitter. Other works suggested to process

[12] http://www.tweetcongress.org

text streams for a variety of NLP tasks e.g., real-time opinion mining and sentiment analysis in social media (Pang and Lee, 2008), named entity disambiguation (Sarmento et al., 2009), statistical machine translation (Levenberg et al., 2011), first story detection (Petrović et al., 2010), and unsupervised dependency parsing (Goyal and Daumé, 2011). Massive Online Analysis (MOA) toolkit developed by Bifet et al. (2010) is an alternative to the Jerboa package used in this work developed by Van Durme (2012a). MOA has been effectively used to detect sentiment changes in Twitter streams (Bifet et al., 2011).

9 Conclusions and Future Work

In this paper, we extensively examined state-of-the-art static approaches and proposed novel models with dynamic Bayesian updates for streaming personal analytics on Twitter. Because our streaming models rely on communications from Twitter users and content from various notions of user-local neighborhood they can be effectively applied to real-time dynamic data streams. Our results support several key findings listed below.

Neighborhood content is useful for personal analytics. Content extracted from various notions of a user-local neighborhood can be as effective or more effective for political preference classification than user self-authored content. This may be an effect of 'sparseness' of relevant user data, in that users talk about politics very sporadically compared to a random sample of their neighbors.

Substantial signal for political preference prediction is distributed in the neighborhood. Querying for more neighbors per user is more beneficial than querying for extra content from the existing neighbors e.g., 5 tweets from 10 neighbors leads to higher accuracy than 25 tweets from 2 neighbors or 50 tweets from 1 neighbor. This may be also the effect of data heterogeneity in social media compared to e.g., political debate text (Thomas et al., 2006). These findings demonstrate that a substantial signal is distributed over the neighborhood content.

Neighborhoods constructed from friend, user mention and retweet relationships are most effective. Friend, user mention and retweet neighborhoods show the best accuracy for predicting political preferences of Twitter users. We think that friend relationships are more effective than e.g., follower relationships because it is very likely that users share common interests and preferences with their friends, e.g. Facebook friends can even be used to predict a user's credit score.[13] User mentions and retweets are two primary ways of interaction on Twitter. They both allow to share information e.g., political news, events with others and to be involved in direct communication e.g., live political discussions, political groups.

Streaming models are more effective than batch models for personal analytics. The predictions made using dynamic models with Bayesian updates over user and joint user-neighbor communication streams demonstrate higher performance with lower resources spent compared to the batch models. Depending on user political involvement, expressiveness and activeness, the perfect prediction (approaching 100% accuracy) can be made using only 100 - 500 tweets per user.

Generalization of the classifiers for political preference prediction. This work raises a very important but under-explored problem of the generalization of classifiers for personal analytics in social media, also recently discussed by Cohen and Ruth (2013). For instance, the existing models developed for political preference prediction are all trained on Twitter data but report significantly different results even for the same baseline models trained using bag-of-word lexical features as shown in Table 1. In this work we experiment with three different datasets. Our results for both static and dynamic models show that the accuracy indeed depends on the way the data was constructed. Therefore, publicly available datasets need to be released for a meaningful comparison of the approaches for personal analytics in social media.

In future work, we plan to incorporate iterative model updates from newly classified communications similar to online perceptron-style updates. In addition, we aim to experiment with neighborhood-specific classifiers applied towards the tweets from neighborhood-specific streams e.g., friend classifier used for friend tweets, retweet classifier applied to retweet tweets etc.

Acknowledgments

The authors would like to thank the anonymous reviewers for their helpful comments.

[13] http://money.cnn.com/2013/08/26/technology/social/facebook-credit-score/

References

Shane Bergsma, Matt Post, and David Yarowsky. 2012. Stylometric analysis of scientific articles. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 327–337.

Shane Bergsma, Mark Dredze, Benjamin Van Durme, Theresa Wilson, and David Yarowsky. 2013. Broadly improving user classification via communication-based name and location clustering on Twitter. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1010–1019.

Albert Bifet, Geoff Holmes, Bernhard Pfahringer, Philipp Kranen, Hardy Kremer, Timm Jansen, and Thomas Seidl. 2010. MOA: Massive online analysis, a framework for stream classification and clustering. *Journal of Machine Learning Research*, 11:44–50.

Albert Bifet, Geoffrey Holmes, Bernhard Pfahringer, and Ricard Gavaldà. 2011. Detecting sentiment change in Twitter streaming data. *Journal of Machine Learning Research*, 17:5–11.

John D. Burger, John Henderson, George Kim, and Guido Zarrella. 2011. Discriminating gender on Twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1301–1309.

Raviv Cohen and Derek Ruths. 2013. Classifying Political Orientation on Twitter: It's Not Easy! In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 91–99.

Michael D. Conover, Bruno Gonçalves, Jacob Ratkiewicz, Alessandro Flammini, and Filippo Menczer. 2011. Predicting the political alignment of Twitter users. In *Proceedings of Social Computing*, pages 192–199.

Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. 2010. A latent variable model for geographic lexical variation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1277–1287.

Rong En Fan, Kai Wei Chang, Cho Jui Hsieh, Xiang Rui Wang, and Chih Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Katja Filippova. 2012. User demographics and language in an implicit social network. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1478–1488.

Nikesh Garera and David Yarowsky. 2009. Modeling latent biographic attributes in conversational genres. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 710–718.

Daniel Gayo-Avello. 2012. No, you cannot predict elections with Twitter. *Internet Computing, IEEE*, 16(6):91–94.

Jennifer Golbeck, Justin M. Grimes, and Anthony Rogers. 2010. Twitter use by the u.s. congress. *Journal of the American Society for Information Science and Technology*, 61(8):1612–1621.

Amit Goyal and Hal Daumé, III. 2011. Approximate scalable bounded space sketch for large data NLP. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 250–261.

Vasileios Lampos, Daniel Preotiuc-Pietro, and Trevor Cohn. 2013. A user-centric model of voting intention from social media. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 993–1003.

Abby Levenberg, Miles Osborne, and David Matthews. 2011. Multiple-stream language models for statistical machine translation. In *Proceedings of the Sixth Workshop on Statistical Machine Translation (WMT)*, pages 177–186.

Yang Liu, Qun Liu, and Shouxun Lin. 2005. Log-linear models for word alignment. In *Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL)*, pages 459–466.

Ingrid Lunden. 2012. Analyst: Twitter passed 500M users in june 2012, 140m of them in US; Jakarta 'biggest tweeting' city. http://techcrunch.com/2012/07/30/analyst-twitter-passed-500m-users-in-june-2012-140m-of-them-in-us-jakarta-biggest-tweeting-city/.

Diana Maynard and Adam Funk. 2012. Automatic detection of political opinions in tweets. In *Proceedings of the 8th International Conference on The Semantic Web (ESWC)*, pages 88–99.

Felix Ming Fai Wong, Chee Wei Tan, Soumya Sen, and Mung Chiang. 2013. Quantifying political leaning from tweets and retweets. In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Dong Nguyen, Noah A. Smith, and Carolyn P. Rosé. 2011. Author age prediction from text using linear regression. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 115–123.

Dong Nguyen, Rilana Gravel, Dolf Trieschnigg, and Theo Meder. 2013. "How old do you think I am?" A study of language and age in Twitter. In *Proceedings of the AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 439–448.

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. 2010a. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 122–129.

Brendan O'Connor, Jacob Eisenstein, Eric P. Xing, and Noah A. Smith. 2010b. A mixture model of demographic lexical variation. In *Proceedings of the NIPS Workshop on Machine Learning and Social Computing*.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations of Trends in Information Retrieval*, 2(1-2):1–135, January.

Marco Pennacchiotti and Ana-Maria Popescu. 2011a. Democrats, republicans and starbucks afficionados: user classification in twitter. In *Proceedings of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 430–438.

Marco Pennacchiotti and Ana Maria Popescu. 2011b. A machine learning approach to Twitter user classification. In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM)*, pages 281–288.

Saša Petrović, Miles Osborne, and Victor Lavrenko. 2010. Streaming first story detection with application to Twitter. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Hoifung Poon, Colin Cherry, and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 209–217.

Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. 2010. Classifying latent user attributes in Twitter. In *Proceedings of the 2nd International Workshop on Search and Mining User-generated Contents (SMUC)*, pages 37–44.

Delip Rao, Michael Paul, Clay Fink, David Yarowsky, Timothy Oates, and Glen Coppersmith. 2011. Hierarchical Bayesian models for latent attribute detection in social media. In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Luís Sarmento, Alexander Kehlenbeck, Eugénio Oliveira, and Lyle Ungar. 2009. An approach to web-scale named-entity disambiguation. In *Proceedings of the 6th International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM)*, pages 689–703.

Noah A. Smith. 2004. Log-linear models.

Craig Smith. 2013. May 2013 by the numbers: 16 amazing Twitter stats. http://expandedramblings.com/index.php/march-2013-by-the-numbers-a-few-amazing-twitter-stats/.

Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 327–335.

A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. 2010. Predicting elections with Twitter: What 140 characters reveal about political sentiment. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pages 178–185.

Benjamin Van Durme. 2012a. Jerboa: A toolkit for randomized and streaming algorithms. Technical report, Human Language Technology Center of Excellence.

Benjamin Van Durme. 2012b. Streaming analysis of discourse participants. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 48–58.

Yi Yang and Jacob Eisenstein. 2013. A log-linear model for unsupervised text normalization. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 61–72.

Tao Yano and Noah A. Smith. 2013. What's worthy of comment? content and comment volume in political blogs. In *International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Tao Yano, Dani Yogatama, and Noah A. Smith. 2013. A penny for your tweets: Campaign contributions and capitol hill microblogs. In *Proceedings of the International AAAI Conference on Weblogs and Social Media (ICWSM)*.

Faiyaz Al Zamal, Wendy Liu, and Derek Ruths. 2012. Homophily and latent attribute inference: Inferring latent attributes of Twitter users from neighbors. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pages 387–390.