Master thesis
Computer Science

Radboud University

User Location Detection
Metadata based classification

Author:
Brian Westerweel
s4802594

First supervisor/assessor:
Prof.dr.ir. Arjen P. de Vries
`arjen@cs.ru.nl`

Second assessor:
Martha Larson
`m.larson@let.ru.nl`

March 29, 2021

Abstract

In this research we set out to detect the location of users of chat applications based on the metadata of their messages. Related work shows that location detection is possible based on the activity profile of the users and then especially using the night inactivity. It also shows that the location of a user can be detected based on the strength of their relationship with respect to their social network.

We contribute to earlier research by focusing on the daily routine of a user's activity profile, other network features besides the strength of the relation between the users and we investigate how the combination of daily routines and network features can help the model to better detect the user's location.

We found that nightly inactivity is a very strong predictor of the location and it is possible to predict the user's country location with 10.86% accuracy, almost doubling on a random guess with the 18 countries in our data set. Our network features include tie strength, density, average start sleep time and number of contacts. These features help the model to detect better than with a random guess with a total accuracy of 7.55%. Combining the two independent set of features improves statistically significant upon both individual models to an accuracy of 11.85%.

Contents

Chapter 1

Introduction

Social media and instant messaging services take a more and more prominent place in every day life. Applications like Facebook and WhatsApp provide means for people to stay in touch with each other regardless of the distance between them. Besides allowing people to communicate with those who are in close physical proximity, social media also give the possibility to connect to people on the other side of the world.

According to Kemp et al. (2020) the number of people using the internet worldwide has grown with 7% last year, to a total of 4.54 billion internet users. Kemp et. al. also found that there were 5.19 billion mobile phone users and 3.80 billion social media users in January 2020.

Of all mobile phone users 89% of them use a chat application or a social networking application each month. On average the global population spends 2 hours and 24 minutes on social media each day.

As social media is widely used by the public, coorperations and governments are investigating ways to use it to their benefit. Several studies, Barwise and Strong (2002) and Barnes and Scornavacca (2004), demonstrate that the use of social media and mobile phone applications such as Facebook are increasingly interesting to publish marketing campaigns. A growing number of companies use messaging services, for instance WhatsApp, to offer customer support. Besides using social media for marketing purposes more recent research has proven the use of social media for social importance. For example, Paul and Dredze (2011) have monitored public health using social media and Mandel et al. (2012) have tracked hurricanes by performing sentiment analysis of Tweets.

Some social media applications allow users to provide their locations. These options include setting a ”home” location in the user profile. Applications like Twitter also offer users the option to provide their current location for each Tweet they send.

Both ways of a user providing its own location are used to model public phenomena such as hurricanes or disease outbreaks. The number of Twitter messages regionally peaks when a certain type of disaster is about to hit that area as the study of Mandel et al. (2012) showed.

The research of Paul and Dredze (2011), on the other hand, is focussed on monitoring health related Tweets. One of the tasks they accomplish is to localize illnesses to geographic regions. Evacuation warnings and Amber alerts can also profit from having a location for the target group. For example, it is undesirable to ask people to evacuate if they do not live in the area at risk.

Public businesses also have great interest in location data. With the location data companies can improve the targeting for marketing campaigns. Service providers offer new services to their customer or added value based on the customer's location as Barnes (2003) found.

A major challenge is the fact that location annotated data is rare. Backstrom et al. (2010) found that about 6% of the Facebook users in the United States have entered a home addresses in their profile. According to Sloan and Morgan (2015) 41.6% of the Twitter users have location services enabled. However, only 3.1% of these users have one or more geotagged Tweets. Cheng et al. (2010) have randomly sampled over 1 million Twitter users and found that only 26% published their home location on a city-level in their profiles. Only 0.42% of the Tweets in the dataset was tagged with a geo-location. Not all of the geo-locations within the data are actual real world locations. Since people are free to provide their own (potentially imaginary) locations, countries like Narnia or Middle Earth occur in the data. The percentage of users that disclose their personal information online is decreasing as Stutzman et al. (2013) found in a research amongst Facebook users. In a study of online discussed topics Choi et al. (2017) found that people in general are becoming more aware about their online privacy. Therefore, it is likely that the percentage of (valid) home location that is publicly available will drop even further in the future.

With instant messaging services geotagging is often not possible. Some currently popular applications allow users to share their current location with others for temporary movement tracking. However, those applications are not publicly linking the user's location to messages they have sent there. This way it is impossible for external researchers to use this data for their own research.

The purpose of this research is to determine the user's location using the meta data of an instant messenger application. In previous research either the user's online activity or his or her social network have been used for this purpose.

The contribution of this research is to combine these two components.

Until now, similar research (e.g., La Morgia et al. (2018) and Webb (2017)) relies on the periods people sleep, thus are inactive for a longer period of time, to determine where that person lives. Those researchers do this by matching the sleep inactivity to the different time zones. They make nightly activity profiles for the different time zones and then look which time zone profile is closest to that of the user. Our research does not only focus on the sleep inactivity but also on when users can post throughout their waking hours.

To find differences between the daily routines of users we artificially let their days start at the same moment. This way we set out to differentiate between countries based on the daily routine of their inhabitants. After we have a data set in which each user starts their day at the same moment, we can also easily remove the night inactivity from the activity profiles. Removing the night inactivity enforces the model to only focus on the daily routines of the users.

The daily routine is interesting to investigate because it is a different focus compared to earlier focus. It is also interesting to see if the day activities of users hold any new information on its own and as Webb found out prior information such as prayer times can improve the location detection of users.

The main research question is:

> *What is the impact of combining network based features with activity based features compared to the individual features on the detection of a user's location?*

To further specify we investigate what the impact of daily routines (i.e., user activity profiles) is on the location prediction. For this we answer the sub question:

> *What is the impact of the daily routines of instant messenger users on the prediction of their physical location?*

We do not only investigate the impact of different daytime activity patterns on location prediction, but we also consider the social network of a user. The goal is to acquire extra features solely based the user's contacts to improve the prediction scores. This introduces the final sub question:

> *What does the social network of a user contribute to the prediction of his or her location?*

My research about the location detection helps the police approximate the physical location of a criminal user and narrow down potential suspects because the police observed the same transfer of human interactions from the physical world to the online world. Therefore, they need to elaborate their online investigation capabilities so they can still locate and identify criminals. The online world also provides criminals with the tools to communicate privately without the police being able to listen in. This means that the police wants to get the most out of the information they can request at companies (i.e., the metadata of interactions).

The police only can requests this information when they have proven to a judge that there is enough suspicion of criminal activity at companies. This proof has to comply with lots of legal and ethical rules before a judge gives the permission to request metadata at companies. Government agencies are constantly monitored for compliance with all legal and ethical rules.

An important note is that this research is done in collaboration with the Dutch police and this research can only potentially be used when they have the permission of judges and legally obtained the necessary metadata. To privately use this research is considered invasion of privacy and is illegal, thus should not be done.

Chapter 2

Related Work

Location detection for forums and chat applications has been the focus of a lot of research. Through the years, several methods of location detection have been utilized from entity recognition methods (i.e., looking in the content of messages for the names of locations) to only using the times of online interactions (i.e., using activity profiles).

Different studies have used different granularities to detect the locations of the users. One research focuses on correctly predicting the time-zone or continent of the users, whereas others want to pin-point the city or even house address.

2.1 Content based

In the last decade, several methods have been published to infer the location of Twitter users based on the content of their Tweets. Within the content based approach two distinct categories of methods can be recognized.

The first category analyses the text for terms related to geographical locations and compares those terms to a gazetteer (an external knowledge base). Examples of this include the works of Fink et al. (2009) and Amitay et al. (2004), where they extracted postal code and addresses from the contents of web pages and retrieved the corresponding geographical locations from gazetteers.

The second category of content based models uses probabilistic language models. For example a Bayesian inference model to predict locations based on the labels of photos taken by Flickr users as Serdyukov et al. (2009) proposed. Based on the tags (i.e., user chosen text labels) of the photo they are able to distinguish between St. Petersburg, Florida, USA and St. Petersburg, Russia. To further improve their model they also combine it with the aforementioned gazetteer method.

2.2 Activity profile based

Besides analysing the contents of a user's posts, it is sometimes possible to make a good estimate of a user's geolocation solely based on a user's activity profile. Such an activity profile encapsulates when the user is posting online messages or chats with another user. With this method only the timestamps of the messages the user sends or receives are needed.

An activity profile based way to retrieve the geolocation of users on the dark web was found by La Morgia et al. (2018). They create user profiles by binning the observed period in multiple hourly bins (i.e., bins representing an hour 1.00 p.m. - 1.59 p.m.). For each bin the number of sent messages are counted and put in the bin. Thereafter all bins representing the same hour are accumulated. So, all bins representing the 1.00 p.m. - 1.59 p.m. of different observed days are put together. To normalize the data, the number of messages in each of the accumulated bins is divided by the total number of days in the observed period. This results in the average messages per hourly bin per observed user. To create a generic profile the average messages per hour of all users in the same locations are summed up and divided by the number of users in that location, resulting in the average number of messages per hour for that location. These locations can be of various granularities ranging from zip codes to continents.

La Morgia et al. (2018) have created generic profiles for different timezones based on Twitter users with at least 30 posts and a public location in their profile. They found that modelling the moments users go to sleep resembled a Gaussian. The distance between the user's individual sleep Gaussian and the Gaussians of the generic profiles for the different countries can be calculated. The lowest distance will then be the guess for the country label of the user.

Another model that predicts the location of Twitter users is introduced by Webb (2017). This model is similar to the work of La Morgia et al. (2018) because Webb also works with the inactivity of the observed users. However, Webb created a User Travel Detector which determined when a user was away from its home location. The locations that were designated as travel locations were excluded in his data sets. This detector works by noticing a shift in the user's sleep pattern. Combining the sleep pattern, prayer times and the Travel Detector, Webb was able to predict the location of Muslim Twitter users with an average error of approximately 100 miles. This means that the model is able to predict the location of an identified Twitter user's location within 100 miles of that user's actual location. When adding prior knowledge about specific cultures Webb showed that it is also possible to identify if a user belongs to that culture.

2.3 Network based

Besides content based approaches it is also possible to infer information about specific persons through the network they partake in. For example, it is possible to predict undisclosed information of social media users using released social network data, as is shown by Lindamood et al. (2009). The researchers found out of which sport clubs users are subscribed to, or what their political preference is. When those users want keep their unpublished information to themselves, they should reduce both the information they post about themselves and their connections to other users. The second best option would be to publish as little information about yourself as possible.

How closely related two persons are, is also known as tie-strength. This concept was first introduced in "The strength of weak ties" Granovetter (1983). In his paper he proposed two types of ties namely strong and weak. Strong ties are typically family and close friends, whereas weak ties often are distant relatives.

Using tie strength Mcgee et al. (2013) propose an algorithm to infer the geographical location of a Twitter users based on their connections in the network (i.e., their followers and the people they follow). Their method was able to successfully predict the location of targets, with an average error of 21 miles. To predict the exact locations rather than just relative locations the links in the network need to have a known location. In general they found that the more followers someone has, the farther away the followers tend to be. If the tie strength increases then the geographical distance between the entities decreases. An example of the idea behind the tie strength distance relation can be found in Figure 2.1. Mcgee et al. mention further research could be done by enriching their model with other factors (i.e. word usage between users).

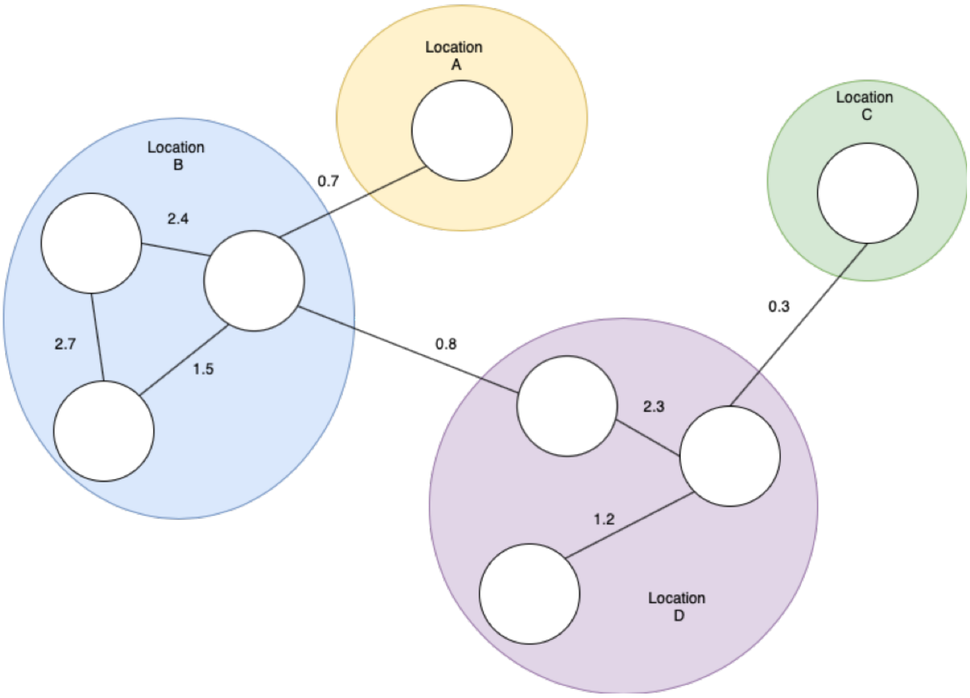

Figure 2.1: Example a network where the tie strength between nodes is displayed. It shows that the higher the tie strength between nodes, the more likely they are to be at the same location.

The data becomes more precise about the geography and social relations the more time people spend online as Backstrom et al. (2010) show. They found the relation between location and friendship using Facebook data. Using those features they propose a method to predict the location of a user using a sparse set of located users (i.e., users that have provided their location). This method outperforms IP-based geolocation algorithms and can scale efficiently to process an almost limitless network.

Chapter 3

Methods

The related works show that multiple types of methods exist that are capable of detecting the users' location based on metadata. These proven methods leave space to improve and expand upon. This research sets out to accomplish the following things.

First we want to explore the potential of the daytime activities of a user to detect its location. This is a new focus compared to earlier research, where they match the sleep pattern to the average sleep pattern of a region (e.g., a country or time-zone), as we will look at the users' activity rather than their inactivity.

In this thesis we investigate how much the daytime pattern of a user contributes to the detection of its location. For this we conduct experiments in which we have removed the part where users sleep from the data.

Secondly, the network based related works focus on using the tie strength between users to detect their location. We would like to investigate what the impact of other network properties is on location detection. These properties are the density of the sender's direct contacts (i.e., how well the senders direct contacts are interconnected), the total number of contacts, the average time (hour and minute) the contacts go to sleep and the average tie strength between the sender and its contacts.

With the average time the users' network goes to sleep we hope to give the model some knowledge about the location of the users' network. Studies showed that the tie strength can determine the relative location of a users compared to their network. With providing the average sleep time we set out to predict more absolute locations for the users. In order to determine the relative position to the network the model needs to have access to the average tie strength.

By providing the total number of contacts we give the model knowledge about the size of a user's network and can find out if the network size is a predictor for the user's location. As we provide information about the density of a user's network we provide information about the what part of the user's network also knows each other. As Mcgee et al. (2013) shows that the number of followers correlates to the location a user lives we set out to find if the aforementioned features are strong predictors for the user location.

Lastly, both the activity based method using the night inactivity and the network based models using the tie strengths have proven to detect the users' location. However, those methods used either the tie strength or the activity patterns to detect the users location. Since both methods contain information that has not been used in the other method, we investigate how combining those two distinct methods improves the location detection.

3.1 Data

For this research a collection of chat messages is used. There are two data sources. One is a collection of messages between actors with a time stamp, the other is a set of actors that have been assigned geographical locations. The actors that have assigned locations can be matched to the collection of messages between actors. For all users the most frequently observed location is used as their location label in these experiments.

The location labels are a combination of the province and the country of a user. For most of the experiments we use the country part from the provinces and use those as the location label rather than the provinces.

The collection of messages contains anonymized identifiers for the sender and receiver of a message and the half hour bin for the date the message was sent. The half hour bins are binary. This means that when there are multiple message from the sender to the same receiver there will be no new entries for that bin. Thus the binary variable only indicates if there was any activity at all. By aggregating the binary variable we can indicate how many user's the sender has contacted in that time slot.

3.1.1 Preprocessing

In pre-processing the following transformations are applied to the data. First, the time stamp field has been used to create date, day of week, hour and minute fields.

Then the entries for specific half hour bins are aggregated by sender and counted. Based on the total amount of bin entries (i.e., how many distinct active half hours a user has in total) a cut is made. We decide to drop the users that have less than 30 entries because La Morgia et al. (2018) found that a more comparable activity model can be build from users that have sent at least 30 messages.

We acknowledge that a user can send multiple messages to the same receiver in the same half hour and it still has one entry for that specific bin. Therefore, users can be more active than it appears from the binned values. However, the idea of the model from La Morgia et al. (2018) is that at least 30 data points are needed for a proper model.

In contrast to Webb's work this research is not using a likelihood formula but only uses the user's actual activity. This method achieved similar results as Webb's method as shown by La Morgia et al. (2018). Rather than estimating the activity for each user we can simply use the observations in the data set. The benefits are a lower complexity of the model (only some configuration is needed) and a higher runtime efficiency.

3.1.2 Sampling technique

During the data exploration phase we noted that there was a large class imbalance. When we trained an early model on this data we verified that the model applied majority voting. If there are different labels that have small difference between the observations, then the model would predict that all those observations belong to the label to the largest label (i.e., the label having most observations in the training set). Therefore, the model would not utilise the small differences between the labels to differentiate between them

To combat the imbalance problem we create a balanced training data set. We want to have each label in our data set to have 200 entries because with 200 entries we are not putting all data points of the smaller labels in, instead of duplicating the data points. The larger labels can also be down-sampled to 200 data points without issues.

We first split the data in training and test set. Then we determine in the training set how many data entries each label has. If a label has less than the 200 entries, we use random oversampling to create 200 entries for that label. With oversampling elements from the original (smaller) data set are picked to create a data set of the desired size Ling and Li (1998). Therefore, some elements can occur multiple times in the re-sampled data set.

If a label has more than 200 entries in the training set, we use random undersampling without replacement to ensure that there would not be a majority in the training samples. Politis (2003) shows that undersampling can be used to improve the models performance when the data is unbalanced. Random undersampling without replacement creates a data set of unique elements selected from the original data set.

After the method has created 200 samples for each label in the training set the re-sampling is done and majority voting is counteracted.

When the data is split after the re-sampling technique is applied, the result can be that there are data points in the test set that also occur in the training set. Thus, the oversampling performed before the split can create trivial problems for the model causing the model learn less from the data. To prevent this problem we choose to re-sample the data after the splitting into training and test sets. This method ensures that none of the samples in the test set occur in the training set.

3.2 Underlying model

The type of model we are training in this research is a random forest as described by Pavlov (2019). We choose this model because it is capable of handling large data sets, can deal with multi-class classification problems and its decisions can be explained to others where needed. Lastly, it also is resistant to overfitting (i.e., focussing on the specific training data set instead of generalizing to better predict unseen data points).

In our case we try to determine what location label belongs to a user. In this research we deal with a multi-class classifcation problem, since we have more than two potential labels for each user. Decision tree based models can handle multi-class classification problems, thus such a model a suits our needs perfectly.

We do this research in collaboration with the police and they prefer a model for which it can be explained why it made certain choices. This means that more black box solutions as machine learning techniques are less desirable as the underlying concepts and decsions are harder to explain.

Random forests are also quite robust against overfitting which makes it a good choice for a multi-class problem. With this model we do not have to provide prior knowledge about class distributions, it also works well with the categorical labels and we do not have to normalise the data.

3.2.1 Random forest classifier

A random forest is a classification model that is created by combining many decision tree models.

A decision tree creates a structure where at each step in the process one binary question is asked to figure out the label of a data observation. The process this model performs is comparable to the children's game *"What's their name?"* in which you get to ask consecutive question to eliminate as many options as possible. The goal is to ask as little questions as possible to get the answer the fastest.

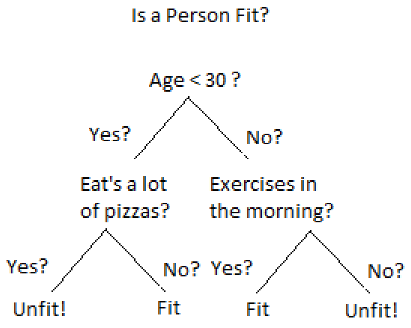

Figure 3.1: Example of a decision tree

Consider the example of a decision tree in Figure 3.1. Here the goal of the classifier is to determine if a person is fit. Each data observation travels through the tree and arrives at an endpoint (i.e., result at the bottom of the tree).

Let us consider the author as a data observation. I am 27 years old, do not exercise every morning and do not eat a lot of pizza. At the first step I follow the left branch since I am aged below 30. Then at the question if I eat a lot of pizza I go to the right, but since that is not the case. The conclusion of this decision tree is that I am a fit person.

Real life situations have a lot more variables to be taken into account. A decision tree can easily adapt to that by simply adding more questions.

A problem with a decision tree is that it can optimize itself to find the most occurring label in the data. The tree loses the focus on the minority class and will not try to label that correctly. It is possible to counteract this when an imbalance in the data set is known by providing the decision tree with this prior knowledge.

Although, one can imagine that the decision tree in Figure 3.1 can be build in different forms while achieving the same results. A random forest is a combination of a large number of decision trees which are all build up differently so that other classes can be labelled earlier than others. Also shifting some variable decisions can lead to whole other results in combination with other parameters. By combining different decision trees the result at the end is more reliable as it is not one tree deciding everything. This also helps against overfitting to the training data, without providing the model with prior knowledge.

3.3 Metrics

In our research we use several metrics to evaluate our results of the experiments.

Accuracy and F1. First of all we use the accuracy to determine how often our model detects the correct location.

When making predictions there are four outcomes that can occur. The given answer can be predicted positive and actually be positive, this is also called true positive. Then the model can predict an answer to be negative and it is actually negative, this is also called true negative. Furthermore the model can predict an answer to be negative while it actually is positive, we call this a false negative. The last option is an answer that is actually positive but the model predicted to be negative, this is also called a false positive.

Then we calculate the accuracy by dividing the true positives by all the positive predictions, either false or true positive.

$$acc = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{3.1}$$

Combined with the accuracy we also calculate the F1-score for the model. Where the accuracy is all about how many of the detected elements are correct (i.e., true positives), the F1-score also considers the elements that are detected with a false label but should have the true label (i.e., false negatives). Both the true positives and false negatives can be grouped together as all the relevant elements. The F1-score equals the mean of the accuracy and the total amount of relevant elements.

$$F_1 = 2 * \frac{acc * recall}{acc + recall} \tag{3.2}$$

The recall is the percentage of all the correctly predicted answers of all the correct predictions that could have been made.

$$recall = \frac{True\ Positive}{True\ Positve + False\ Negative} \tag{3.3}$$

To accommodate the F1-score for a multi-class classification problem we use a weighted averaging. Therefore, the F1-scores are calculated for each class. These scores are then averaged with weights to account for potential label imbalance.

P-value. To determine if the new model outperforms the baseline we calculate the p-value using the chi-squared test, following Zibran (2008), by comparing the classification results between the baseline and the altered model. The null hypothesis is that the altered model performs the same as the baseline model.

We choose for the common significance threshold of 95%, meaning that the p-value should be below 5% to indicate statistical significance. This assumes that in 95% of the cases the change in result is not correlated to the change made in the experiment.

Effect size. Finally, we calculate the effect size of the new model compared to the baseline model. We calculate the effect size with Cramérs V (see Formula 3.4) which works well with the chi-squared test results.

$$V = \sqrt{\frac{\chi^2}{n * min(r-1, c-1)}} \tag{3.4}$$

In this formula χ^2 is the squared result of the chi-squared test, n is the total amount of observations in the chi-squared table. The $min(r-1, c-1)$ references to the freedom there is in the observations, where r and c are respectively the rows and columns in the table. Since our table is a contingency table of two rows by two columns we have a domain of freedom of 1.

Where the p-value can indicate a positive or negative correlation, the Cramérs V indicates how strong the effect of the alterations are. To interpret the Cramérs V we use Cohen's references Cohen (2013). These references can be found in Table 3.1.

Effect size	V
small	0.1
medium	0.3
large	0.5

Table 3.1: Cramérs V references as proposed by Cohen (2013) for a domain of freedom of 1.

3.4 Activity based

To answer our first sub question we train a model that predicts the locations of users based on their activity profiles. Therefore, we start by creating an activity profile for each user.

3.4.1 Activity Profile

An activity profile is created based on the posting rate of a user. The posting rate depends on the activities that the user participates in. In general an activity profile resembles the user's average activity during a 24 hour period. An activity profile shows how likely a user will be active for given times throughout the day. An example of an activity profile can be seen in Figure 3.2. La Morgia et al. (2018) and Webb (2017) both use an activity pattern or user posting rate to predict the user's location.

In this research the activity profile is created by dividing a 24 hour day into 48 thirty minute segments. All messages within a time period are grouped together. The total messages per thirty minutes are divided by the amount of days between the first activity of the user and its last. Lastly, the amount of messages per time period is divided by the total amount of messages sent by that user. Resulting in the percentage of the total messages sent. This normalization step ensures that the model is not biased towards the most frequent users.

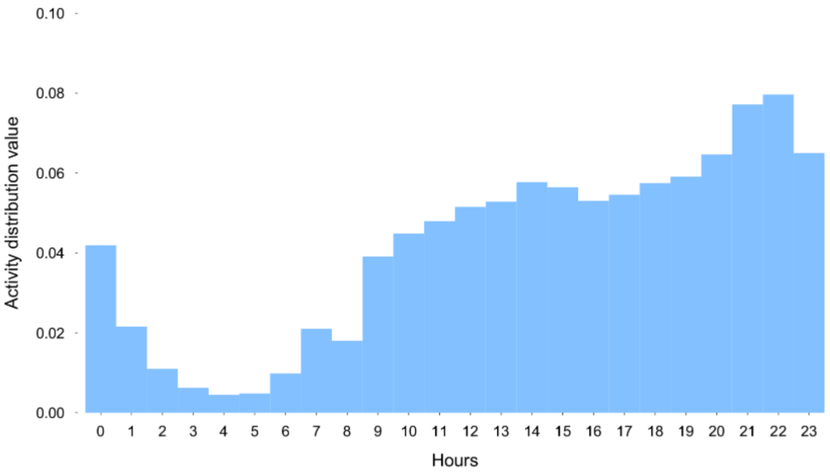

Figure 3.2: Example of activities binned per hour

Webb (2017) uses the activity patterns to look for inactivity on messaging applications during class specific real world activities. These inactivities are used to label the user to that specific class. Webb creates a model to predict the city the users live in, contrary to La Morgia et al. (2018) who use a time-zone granularity for prediction. Where Webb also exploits prior knowledge of class specific activities La Morgia et al. only use the sleep of the users to detect their location.

3.4.2 Night Inactivity

Sleep is a strong indicator for the time zone and even the location of a person, as Webb (2017) and La Morgia et al. (2018) have found. Both authors found that their models learned to match the times people were inactive to the middle of the night hours of their location labels.

In our baseline model we create an activity model with the binned messages as we find them in the data. For each user we calculate the percentage of total messages are sent in each bin (the sum over the averages of all bins equals one).

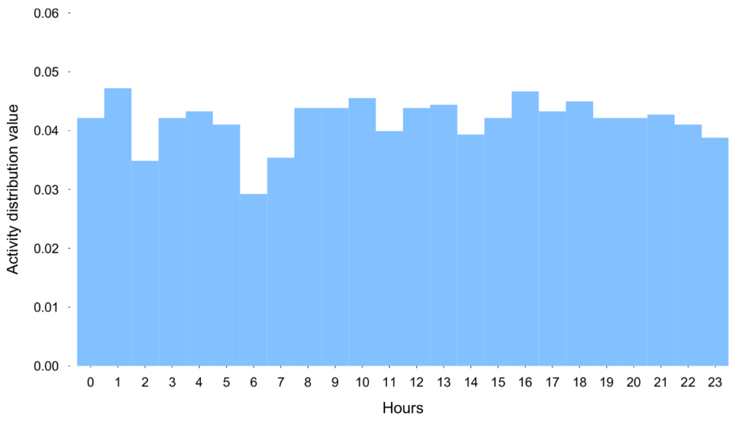

Figure 3.3: The average activity bins of all users gives a plateau-like graph when the start moment of the days is not aligned.

Imagine when the data would be sampled from all time zones equally, then the activity averaged over all users would look similar to Figure 3.3. This data still contains the night inactivity, but shows no clear peaks or valleys, even though each user is accounted for in creating this Figure has a clear inactivity period.

Our baseline model picks up the different night inactivities and uses them to detect the users location. For this model all users in the data still have their night inactivities scattered over the day; for example, Dutch users might start their night inactivity around 23.00 UTC and the east coast of the United States might start their inactivity around 05.00 UTC.

We want to research the effect of the times the users are active. To ensure that the model does not implicitly use night inactivities for the users, we will remove the inactivities from the data.

We did not go through all data entries manually to find the start of the night inactivity. Instead we have developed an algorithm that finds the start times of the night inactivity for us. We assume that the users are sleeping during the large (7 hour) inactivity periods, and refer to the developed algorithm as our sleep detector.

Sleep detector

The goal of the sleep detector is to detect the night inactivity of the users. We define the night inactivity as a period of 7 consecutive hours with the least amount of activity for that user.

By identifying the night inactivity we can exclude the night period and let the model focus on the daily routine of the users. Since the related work section already shows that the night inactivity is a strong predictor for the locations of users, we want to investigate what the daily routine adds to the night inactivity. We test the importance of the daily routine by excluding the night inactivity from the activity profile. Earlier research only considered the activity pattern as a whole and does not make a difference between the night inactivity and the day activity.

The shift in start moment of the night inactivity is an intuitive predictor as the night is not happening globally at the same moment. Therefore, we are curious what information is conveyed in the activities during the day. In this research we mainly want to see if inhabitants of different countries have different daily routines (i.e., in Spain they have a siësta and are assumed to sleep during the day) and we can use these routines to predict the users locations. Webb (2017) shows that adding prior knowledge (such as prayer times) can improve the performance of location prediction models.

We hope to find that the daily routine of users on its own contributes to determining their location. We expect that it is possible to detect the users locations based on their daytime activity pattern. This can also create options for further research to distinguish between different user groups (e.g., students and office workers).

First we build an activity profile as described in Section 3.4.1. Thereafter, we take the sum of 14 consecutive half hours (i.e., 7 hours) and assign the value to the first half hour of the used period. As we made the assumption that everyone is inactive for seven consecutive hours knowing the start moment of the inactivity period is sufficient. The start of the sleeping period is calculated using the averaged and normalized binned data of the users. Hence, the start of the sleeping period for a given user is the time they go to sleep on average.

At last the sleep detector returns the time with the lowest activity in the next seven hours. Thus, when the algorithm is applied to all users in the data set each user has the time assigned to the start sleep variable it starts their inactivity period.

Algorithm 1: Sleep detector algorithm

```
Result: The moment the user starts his or her night inactivity
list_of_start_moments;
for start_moment in (all_half_our_segments) do
    n_activities = aggregate activities of the next 7 hours
      list_of_start_moments[start_moment] = n_activities
end
return start_moment with lowest n_activities
```

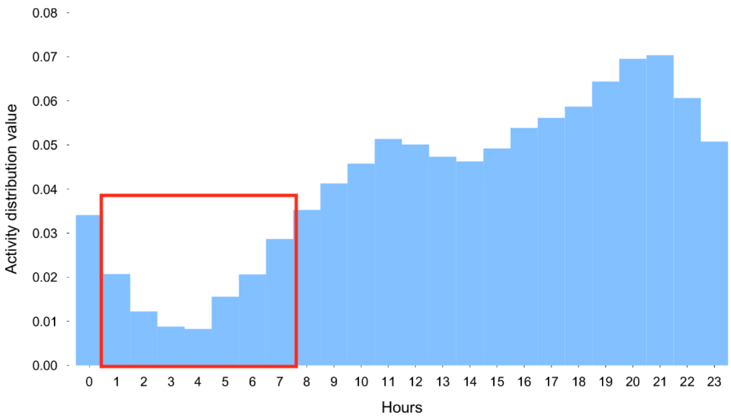

Figure 3.4: An example where the red outline shows the night inactivity as our sleep detector would find it.

The example above shows the binned data aggregated hourly. When we try to find the start sleep moment we will use 7 time periods instead of 14 (i.e., considering the different period length). The red outline in Figure 3.4 gives the period where the sleep detector finds the lowest activity. For this user 01.00 UTC would be returned as the start sleep time.

Data set time alterations

With the estimated moments for each user when they start their night inactivity, we can create data sets to identify the impact of daily routines on the location detection.

We create two additional data sets. One where all the start sleep times are aligned (the offset dataset) and another (the daytime dataset) where we, besides aligning, also removed the night inactivity for all users.

For the first additional data set we artificially let all users start their night inactivity at the same time. We shifted the activity model of each user so that they started their night inactivity at midnight (00.00 UTC). With our assumption that everyone sleeps for seven hours they all start their day at (an imaginary) 07.00 UTC. After aligning the starting points of all sleeping periods the plot of all bins of all users looks more like Figure 3.2. We call this data set the *offset data set*.

To focus on the daily routines as described before we remove the night inactivity for all users to create a second data set. The challenge is that our model cannot know which inactive hours we have removed for the users. The removed hours can also tell the model what hours can be indicative for the location detection. Therefore, we take the *offset data set* and remove all the bins containing the hours 00.00 UTC to 07.00 UTC. As a result we obtain a data set where we have the consecutive bins where the first bin contains 07.00 UTC and the last bin contains 23.30 UTC. This data set is called the *daytime data set*.

3.5 Network based

As Backstrom et al. (2010) and Wiese et al. (2015) have concluded, it is also possible to determine user locations from the location of their friends. They rely on the tie strength between users. However, tie strength is an attribute that is hard to directly define through just online interactions. Several studies, including Granovetter (1983) and Karahalios et al. (2016), conclude that the interaction frequency is a good proxy for tie strength.

When considering only contact frequency, Wiese et al. (2015) found that frequency alone can give the wrong picture about the data as strong ties. People that have very close ties might have a lower interaction frequency as they also communicate by other ways. To encapsulate the lower frequency close ties can have, they rather use the time between the the first interaction between users until the last observed one.

Tie strength not only consists of a frequency, or duration as Granovetter discovered. Therefore, we combine the frequency aspect and the duration aspect of tie strength. Since we cannot interview our users to verify the estimated tie strengths, these two aspects are an assumed proxy for tie strength.

For the network based experiments we use the following features: *the user's total number of contacts, the tie strength between the user and its network, the user's network average start sleep time* and *the user's network density.*

We consider only users in the *200+ data set*. This is kept the same as the activity based experiments to make sure we can make a fair comparison between all the experiments.

Tie strength

Tie strength is not a factual variable of two users but rather an objective interpretation of the relationship between them.

During early experiments we noticed that keeping the tie strengths between all users conveys too much information about the test set. As we do this research to examine approaches for real life problems, keeping all tie strengths was too dissimilar from the real life situation we need to operate in. Therefore, we approximate an average tie strength between users and their networks. We first calculate the tie strengths between a sender (i.e., the user) and all its recipients.

The tie strength between a sender and a recipients is calculated by counting all the times the sender sent a message to the recipient. The total number of messages is divided by the number of days between the first and last moment of observed contact. The result is referred to as tie strength in this research.

To get the average tie strength between a user and their network the tie strengths of all recipients of the user are summed and divided by the amount of recipients.

Density

Besides tie strength we also decided to use the density of the network as a feature. In context of this research a network is considered the user (i.e., sender) and all of its recipients. Possible relations between the recipients are also a part of the considered network. However, relations between one of the recipients and a user that is not directly contacted by the original sender is not part of the considered network.

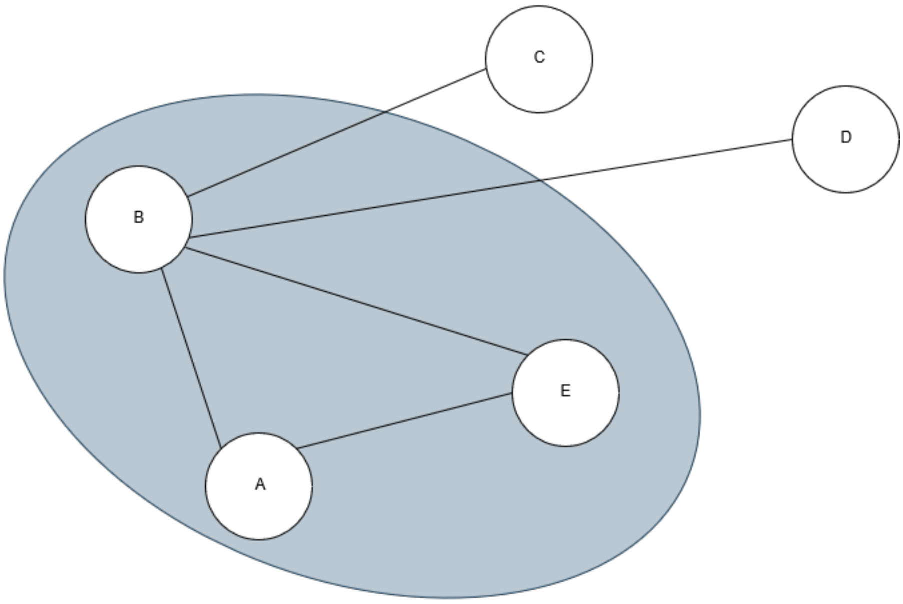

Figure 3.5: An example of the sub-network of which the density will be calculated. We calculate the density for node A. Then we consider nodes B and E since they are direct contacts of A. However, C and D and their connections are not considered since they are no direct contact of node A. The relations between node B and node E are considered in the density calculation as it is between the direct contacts of node A.

To calculate the network density for each user we first construct a network of the user and its recipients. Then the density of this sub-network is calculated. The result is attributed to the user as the density variable.

The density is defined as the actual connections in a network divided by the potential connections.

$$density = \frac{Actual\ connections}{Potential\ connections} \tag{3.5}$$

Where the potential connections are calculated as follows:

$$Potential\ connections = \frac{n * (n-1)}{2} \tag{3.6}$$

The n in this equation is the total number of nodes in the network.

In the network of Figure 3.5 there are 5 nodes. This leads to $\frac{3*(3-1)}{2} = 3$ potential connections. Hence, the density of the network is $\frac{3}{3} = 1$.

For the considered network (i.e. only taking nodes A, B, E and their edges into consideration) there are 3 nodes giving $\frac{3*(3-1)}{2} = 3$ potential connections. With the potential connections known we come to a density of $\frac{3}{3} = 1$.

Chapter 4

Experiments

This research consists of experiments exploring both the aforementioned time series problem, the network problem and a combination of the two.

4.1 Activity based experiments

To determine the impact of daytime activity on location detection the following experiments are performed. Overall we expect that it is possible to detect the user's location based on its daytime activity.

First we create two baseline models. Both are trained on the same data set that had no time alterations. The difference between them is the location labels. The first baseline model has the users' provinces as the label where the second model uses the users' countries as labels. We decide to use the countries as labels to reduce the total number of possible labels.

The expectation is that the country baseline model performs better than the province model because there are less labels. Another expectation is that the location labels within time zones are harder to distinguish.

We drop all users that have less than 30 recorded activity time slots in preprocessing. On top of that we keep only the labels that have a certain number of users assigned to them, so our model can make a better approximation for each label.

We assume that the model for a country will be comparable to that of a user only on a large scale. Therefore, a cut off of at least 30 users would suffice. We chose 200 users as a cut-off because we found a jump from slightly over 30 users to labels with 200+ users during the data exploration. This leaves us with a total of 18 labels. The data set that contains all users labelled with a country that contains at least 200 users will be referred to as the *200+ data set*.

The next experiment is a model trained on the *offset data set* subset of the *200+ data set.* This data set is now called the *offset 200+ data set.*

With this experiment we expect to see that the accuracy for location detection slightly drops as the information about the night inactivity is no longer available. However, the model still receives some knowledge about what happens during the night inactivity for the users.

In the next experiment we work with the *daytime data set* subset of the *200+ data set*, now called the *daytime 200+ data set.* Again, we expect a small drop in accuracy because we give the model less information to work with. However, we still expect the performance to be well over a random guess.

This experiments show us how well we can detect the user's location using only their daytime activity.

Experiment	Dataset	Measures
Baseline	200+	Accuracy, F1-score
Artificially synchronised start day	Offset 200+	Accuracy, F1-score Chi-squared (vs. baseline) Cramérs V (vs. baseline)
Activity daytime	Daytime 200+	Accuracy, F1-score Chi-squared (vs. baseline) Cramérs V (vs. baseline)

Table 4.1: An overview of the activity based experiments.

4.2 Network based experiments

With respect to the network features we do several experiments to find the impact of the individual features and the effect of all combined features on the location detection. For all the experiments we use the same underlying model and the *200+ data set.*

For the first experiment we take all the features and train a classifier. This experiment will establish a baseline to which we can compare the impact of the tie strength, density and average sleep time features. We use this experiment to establish a baseline for the next network based experiments.

For the second experiment we remove the tie strength from the features used in the previous experiment. All other settings remain the same. Since we expect the tie strength to hold enough information to detect the user's location we expect the performance to drop significantly compared to the baseline model. The significant drop would indicate that the tie strength is an important network feature to detect the user's location.

The third experiment uses all the features except the density of the user's network. We expect the performance to drop compared to the baseline model but to be lower than that of the model without tie strengths.

The last experiment is similar to the last one but with the network's average start sleep time removed instead of the density. All the other settings are kept the same compared to the baseline experiment. Again we expect the model to perform worse compared to the baseline model.

Experiment	Dataset	Measures
Baseline	200+	Accuracy, F1-score
No tie strength	200+	Accuracy, F1-score Chi-squared (vs. baseline) Cramérs V (vs. baseline)
No density	200+	Accuracy, F1-score Chi-squared (vs. baseline) Cramérs V (vs. baseline)
No average sleep time	200+	Accuracy, F1-score Chi-squared (vs. baseline) Cramérs V (vs. baseline)
No total amount of contacts	200+	Accuracy, F1-score Chi-squared (vs. baseline) Cramérs V (vs. baseline)

Table 4.2: An overview of the network based experiments.

4.3 Combining network and activity based models

In Section 4.1 we researched how well a user's location can be detected by solely using his activity model. In Section 4.2 we looked at the impact of the user's network features on itself to the detection of its location. In this section we will explore what both detection methods can contribute to each other.

We will combine both feature sets into one larger set. This set holds for each user its day time data set activity model (i.e., without the night inactivity), the start sleep time of the user and all the aforementioned network feature set. For this experiment we will consider the *200+ data set*.

We expect that this model will score better than the other two model types. Because the network data and the activity models both contain information the other model doesn't have access to. Therefore, combining all information should give the model the opportunity to outperform the other models.

Experiment	Dataset	Measures
Combination model	Daytime 200+	Accuracy, F1-score Chi-squared (vs. activity daytime) Cramérs V (vs. activity daytime)

Table 4.3: An overview of the combining network and activity based experiment.

Chapter 5

Results

This chapter contains the results of all conducted experiments. Each of the experiment sections has their counterpart in this chapter.

The p-values and Cramérs V values for each model are calculated compared to the baseline model. The baseline model does not have the p-value and Cramérs V values, as it does not make sense to compare a model to itself.

5.1 Activity Profile

Table 5.1 below shows all the results of the experiments regarding the activity profile experiments.

Experiment	Accuracy (%)	F1-score	P-value	Cramérs V
200+ data set	15.86	15.28		
Offset 200+ data set	13.67	13.76	$4.11e^{-5}$	0.03
Daytime 200+ data set	10.86	12.11	$1.27e^{-22}$	0.07

Table 5.1: Results of the activity profile experiments.

Our baseline model trained on *200+ data set* contains all user's that belonged to labels that contained 200 or more users, this leaves us with 18 labels. The activity for these users is aggregated and averaged per hour. This model can detect a user's location with 15.86% accuracy which is almost triple the accuracy for a random guess on 18 labels which is 5.5%.

When we align the moments users go to sleep we can see that the accuracy drops with 2.19% compared with the baseline. As the p-value is smaller than 0.05 this drop is statistical significant. However, the effect of this change is considered small according to the Cramérs V value. This is expected since we removed the implicit time-zone information which was in the data.

If we remove the assumed sleep of the users, we see the accuracy drop again with 5% compared to the baseline model. This loss is statistical significant as the p-value is extremely small. Although the p-value indicating a strong significance, the effect size is still relatively small. The model has even less information about the activity of the users as they all had some activity during their assumed sleep.

5.2 Network Analysis

In Table 5.2 the results with respect to the network based experiments are displayed.

Experiment	Accuracy (%)	F1-score	P-value	Cramérs V
Baseline model	7.55	9.01		
Removed tie strength	7.26	8.62	**0.58**	
Removed density	**7.66**	**9.08**	0.86	
Removed average sleep time	7.37	8.79	0.74	
Removed total amount of contacts	7.35	8.66	0.71	

Table 5.2: Results of the network analysis experiments. The removal of several features is done one at a time. All experiments are done with the *200+ data set*.

The experiments we set up not only tell us how well the user's network can help detect its location but also what the impact is of different features.

When we train the baseline model using all the features we obtain an accuracy of 7.55% with an F1-score of 9.01 for a multi-label problem with 18 labels. This means that it is only slightly better (i.e., 1.99% improvement) than a random guess.

If we remove the tie strength from the features the accuracy drops to 7.26%. This change is not significant as the p-value is 0.58, thus $p > 0.05$.

When we remove the user's network density the accuracy improves to 7.66%. Though we observe an accuracy improvement the p-value worsens.

Whereas we remove the average start sleep time of the user's network the accuracy becomes 7.37%. The p-value still indicates that this change is statistically insignificant.

With the removal of the total amount of contacts the accuracy of the model is 7.35% an the p-value is 0.71, thus the change is not statistically significant.

5.3 Combined analysis

Table 5.3 shows the results of the experiments that combine both the activity and the network feature sets.

Experiment	Accuracy	F1-score	P-value	Cramérs V
Activity features + network features	11.85	11.89	**0.03**	**0.02**

Table 5.3: Results of the experiments in which the network and activity features are combined. The model is trained with the *Daytime 200+* data set.

The results show that the model based on both the network as the daytime activity features achieves an accuracy of 11.85% for the *200+ data set.* In other words, we observe an accuracy increase of 0.99% compared to the activity based model for *daytime 200+ data set.* This change is statistically significant with a $p < 0.05$. The effect size is considered very small with a Cramérs V of 0.02.

Chapter 6

Conclusions

In Chapter 4 we conducted several experiments to explore the impact of the user's daily routine on the location detection. This chapter covers the conclusions from the experiments and links them to the proposed research questions.

We started by creating experiments for detecting the user's location based on its activity to answer our first sub question:

> *What is the impact of the daily routines of instant messenger users on the prediction of their physical location?*

We noticed that although the *daytime 200+ data set* gives a worse performance than the *offset 200+ data set* it still accomplishes a 10.86% accuracy which is doubling the score of random guessing. When we add the start sleep indication to the *daytime 200+ data set* the model performs better than the baseline model and even the *offset 200+ data set* based model.

After these experiments we conclude that training a model on the *daytime 200+ data set* outperforms the random guess, but does not outperform the models leverage the night inactivity. So, it is possible to detect a user's location based on his or her daily routine.

We have also explored the options of detecting a user's location by only using its network features. These experiments help us answer the last sub question:

> *What does the social network of an user contribute to its location?*

We find that training a network based model using our features only slightly improves upon a random guess. Although none of the individual features has a significant effect on the model, the average tie strength feature has the largest effect size. We also see that the baseline trained on the

200+ data set with network features performs worse than its activity profile based counterpart. So, we conclude that the model trained on our network features is more accurate than a random guess.

Next we examine the result of the experiment combining the network and activity based models. With this experiment we can answer the main research question:

> *What is the impact of combining network based features with activity based feature compared to the individual features on the detection of a user's location?*

We see that the model trained on the combination of network features and *daytime 200+ data set* activity profile features performs significantly better than the activity profile only model. Which means it also performs significantly better than the network features only model.

Therefore, we conclude that the impact of combining network based features with activity based features significantly increases upon their individual performances. Combining both independent feature sets helps to perform the detection model.

Chapter 7

Discussion and Future Work

In this chapter we discuss several decisions and assumptions that we made during the research. We also point to aspects that were untouched in this research but are interesting to explore in future works.

The first matter we want to discuss is the use of IP addresses to determine geo-locations. Nowadays it is possible to find plenty of website that offer IP to geo-location applications.

However, those online services are not 100% accurate as they rely on other IP services such as the American Registry for Internet Numbers WhoIs service to retrieve information about IP addresses. The online service combine the information of multiple WhoIs service to finalize there conclusion. The problem is that each of intermediate services has a certain error margin.

In recent research Livadariu et al. (2020) showed empirically that not all services contain all IP addresses and different services have different levels of geo-locations labelled to the IP. These different levels are either, continent, country, province or even city. Besides, they found that for 62.85% of the IPv4 and 78.90% of the IPv6 addresses could be provided with both the city level location and the owner of the address. For other addresses higher level locations (e.g., countries or time-zones) are found.

Because the higher level locations need to be matched to other services that can find a more detailed location it provides a lot of room for errors. Adding disagreement between different sources also enables more mistakes to be made.

Combining all factors make this automatic geo-location assignment far from ideal to use it for province or city level ground truths.

When we checked a known IP address it was assigned to be in a city about 20 kilometres from the actual location. In this case both the province and country were correct.

Working with country labels makes the ground truth suffer the least from possible error and can be accepted as ground truth in our research. In future works it could be worthwhile to only accept IP's in the data can be linked to the same geo-location among all services.

However, this will then make the research prone to overfitting on an ideal world as the real world contains the excluded IP addresses. When location detection methods are developed also new IP's or IP's with disagreement about their linked location should be taken into account.

When we first started this research we thought it was interesting to identify user groups within the labels. The goal would be to see if it is possible to user group label specific activity patterns. However, due to the limited time and elaborate experiments for the other sub questions we decided to drop this idea.

Besides, recognizing label specific activity patterns this research can uncover universal activity patterns. An example for this is used by Webb (2017) as he used given prayer times to identify inactivities for religious Twitter users.

For future work it would be nice to see what different user groups can be identified. This can be done with prior knowledge of prayer times, holidays and such. It would also be interesting to see if user groups surface about which we have we have little to no prior knowledge.

Another issue that warrants further research is the skewed data distribution. As described in the experiments, the 5 largest labels contain more than half of the data in the selected set. Therefore, if we want to correctly detect the users belonging to smaller labels we have to use re-sampling methods to balance the labels.

For the smaller labels we used in our research we had to use up-sampling. This means that we have duplicates in the training data for the smaller labels and unique data points in the larger labels. In our case re-sampling to 200 samples per label helped improve the accuracy on the smaller labels, but it lost accuracy for the larger labels.

It would be good to use larger samples for all the labels. For training 200 samples is not that much especially since we did not apply any data augmentations to artificially increase the amount of samples per label. Therefore, future research could benefit from more data for the smaller labels. Options to achieve more data are artificially augmenting data, more sampling for the smaller labels or limiting the models to labels that contain more data points.

Overall, an option to increase the accuracy for detecting the users is by combining detection models for different granularities. Our research directly tried to detect the label of a user.

Future research could try to train a cascade model, where they first train a model that detects the time zone of users. This can be done reliably, as described in the related work section. Thereafter, granularity models can be trained for each time-zone. By training the time zone specific models such a model is forced to focus on smaller details between users to detect the correct country within an time zone. In our research, by focussing on the day daily routine we removed the information about the night and when the night started. This includes the indirect information about the time zone. This can lead to misclassifications because users behave similar activity patterns during the day, but actually live in whole other time zones. The aforementioned sequential activity models can counter this. Because a time zone can be indicated first and the data point can than be passed to the correct time zone specific model. This also can mean that the data does not have to be pre-processed to get rid of the night pattern.

Bibliography

Amitay, E., Har'El, N., Sivan, R., and Soffer, A. (2004). Web-a-where: Geotagging Web content. *Proceedings of Sheffield SIGIR - Twenty-Seventh Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 273–280.

Backstrom, L., Sun, E., and Marlow, C. (2010). Find me if you can: improving geographical prediction with social and spatial proximity. In *Proceedings of the 19th international conference on World wide web*, pages 61–70.

Barnes (2003). Location-Based Services: The State of the Art. *e-Service Journal*, 2(3):59.

Barnes, S. J. and Scornavacca, E. (2004). Mobile marketing: The role of permission and acceptance. *International Journal of Mobile Communications*, 2(2):128–139.

Barwise, P. and Strong, C. (2002). Permission-Based Mobile Background : the Growth of Mobile. *Journal of Interactive Marketing*, 16(1):14–24.

Cheng, Z., Caverlee, J., and Lee, K. (2010). *You Are Where You Tweet: A Content-Based Approach to Geo-locating Twitter Users.*

Choi, H. S., Lee, W. S., and Sohn, S. Y. (2017). Analyzing research trends in personal information privacy using topic modeling. *Computers & Security*, 67:244–253.

Cohen, J. (2013). *Statistical power analysis for the behavioral sciences.* Academic press.

Fink, C., Piatko, C., Mayfield, J., Finin, T., and Martineau, J. (2009). Geolocating blogs from their textual content. *AAAI Spring Symposium - Technical Report*, SS-09-08:25–26.

Granovetter, M. (1983). The strength of weak ties: A network theory revisited. *Sociological theory*, pages 201–233.

Karahalios, K., Gilbert, E., Karahalios, K., and Sandvig, C. (2016). The network in the garden : An empirical analysis of social media in rural life The Network in the Garden : An Empirical Analysis of Social Media in Rural Life. (March):1603–1612.

Kemp, S., Currey, H., and Oxford, L. (2020). Digital 2020: 3.8 billion people use social media.

La Morgia, M., Mei, A., Raponi, S., and Stefa, J. (2018). Time-zone geolocation of crowds in the dark web. *Proceedings - International Conference on Distributed Computing Systems*, 2018-July(Idc):445–455.

Lindamood, J., Heatherly, R., Kantarcioglu, M., and Thuraisingham, B. (2009). Inferring private information using social network data. In *WWW'09 - Proceedings of the 18th International World Wide Web Conference*, pages 1145–1146.

Ling, C. X. and Li, C. (1998). Data mining for direct marketing: Problems and solutions. In *Kdd*, volume 98, pages 73–79.

Livadariu, I., Dreibholz, T., Al-Selwi, A. S., Bryhni, H., Lysne, O., Bjørnstad, S., and Elmokashfi, A. (2020). On the Accuracy of Country-Level IP Geolocation. *ANRW 2020 - Proceedings of the 2020 Applied Networking Research Workshop*, pages 67–73.

Mandel, B., Culotta, A., Boulahanis, J., Stark, D., Lewis, B., and Rodrigue, J. (2012). A demographic analysis of online sentiment during Hurricane Irene. *Proceedings of the 2012 Workshop on Language in Social Media*, (Lsm):27–36.

Mcgee, J., Caverlee, J., and Cheng, Z. (2013). Location Prediction in Social Media Based on Tie Strength. *Citeseer*.

Paul, M. J. and Dredze, M. (2011). You Are What You Tweet: Analyzing Twitter for Public Health. *Int'L AAAI Conference on Weblogs and Social Media (ICWSM)*.

Pavlov, Y. L. (2019). Random forests. *Random Forests*, pages 1–122.

Politis, D. N. (2003). The impact of bootstrap methods on time series analysis. *Statistical science*, pages 219–230.

Serdyukov, P., Murdock, V., and Van Zwol, R. (2009). Placing Flickr photos on a map. *Proceedings - 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009*, (May):484–491.

Sloan, L. and Morgan, J. (2015). Who tweets with their location? Understanding the relationship between demographic characteristics and the use of geoservices and geotagging on twitter. *PLoS ONE*, 10(11):1–15.

Stutzman, F. D., Gross, R., and Acquisti, A. (2013). Silent listeners: The evolution of privacy and disclosure on facebook. *Journal of privacy and confidentiality*, 4(2):2.

Webb, M. R. M. (2017). Infering User Location from Time Series of Social Media Activity. (2017).

Wiese, J., Min, J. K., Hong, J. I., and Zimmerman, J. (2015). "you never call, you never write": Call and SMS logs do not always indicate tie strength. *CSCW 2015 - Proceedings of the 2015 ACM International Conference on Computer-Supported Cooperative Work and Social Computing*, pages 765–774.

Zibran, M. F. (2008). Chi-square Test of Independence. *The Concise Encyclopedia of Statistics*, pages 79–82.