Fine-grained location prediction of non geo-tagged tweets

Mohammad Abboud, Karine Zeitouni, Yehia Taher

Fine-Grained Location Prediction of non Geo-tagged Tweets - A Multi-view Learning Approach

Mohammad Abboud
mohammad.abboud@uvsq.fr
DAVID Lab, UVSQ - Université Paris-Saclay
Versailles, Yvelines, France

Karine Zeitouni
karine.zeitouni@uvsq.fr
DAVID Lab, UVSQ - Université Paris-Saclay
Versailles, Yvelines, France

Yehia Taher
yehia.taher@uvsq.fr
DAVID Lab, UVSQ - Université Paris-Saclay
Versailles, Yvelines, France

ABSTRACT

Geotagged Social Media (GTSM) data, especially geotagged tweets are valuable sources of information for many important applications. Only small portions of geotagged tweets are available (less than 3%). Identifying tweet location is a challenging problem that has attracted the interest of both academic and industry fields. Existing approaches have satisfactory accuracy at country and city level, but fail in locating more precisely the tweets. This paper presents *FLAIR*, an approach for geolocating tweets at finer granularities. Our objective is to predict the tweet location in a well-known and pre-defined area, that is to reduce the distance error between the predicted and real locations. In this work, we propose a location prediction model leveraging spatial model for POIs extracted from a text from one hand, and textual model comparing text similarity between geotagged and non-geotagged tweets, from another hand. We adopt a multi-view learning approach to combine the results of both predictions. Experimental results show that our proposed model outperforms the existing solutions.

CCS CONCEPTS

• **Information systems** → **Information retrieval**.

KEYWORDS

Spatial Databases, Social Media, Natural Language Processing, Location Prediction, Multi-view Learning

ACM Reference Format:
Mohammad Abboud, Karine Zeitouni, and Yehia Taher. 2022. Fine-Grained Location Prediction of non Geo-tagged Tweets - A Multi-view Learning Approach. In *The 5th ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery (GeoAI '22) (GeoAI '22), November 1, 2022, Seattle, WA, USA*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3557918.3565875

1 INTRODUCTION

Social media platforms have experienced a huge boost over the last decade. Users are able to connect with other people through these platforms, develop their online friendships, and share their real-life events with them. They can post their opinions as well as their activities, using text, images, and videos, which makes those platforms valuable sources of information for analysis.

Twitter is one of the most popular social media platforms. It allow establishing non-mutual friendships. Twitter has its own API the *Twitter API*[1] that allows researchers and developers to crawl and analyze tweets. Tweets contain in addition to the textual posts, links to media posted and other useful metadata for analysis.

With the popularity of GPS-enabled mobile devices, Twitter users are now able to share their location voluntarily when tweeting. Furthermore, they are able to mention points-of-interest (POIs) in their tweets, such as names of restaurants, cities, and touristic spots, etc.

Knowing the exact location of tweets can help in monitoring the real world. Many applications can benefit from geotagged text information, a few to list, natural disaster and crime detection [10, 20], health care management [24], marketing recommendation systems [2], and event detection systems [21, 22], etc. Unfortunately only 1 to 3 % of tweets contain geotagging information [18], which makes analysis a hard task within the absence of such data.

Tweets can be categorized as tweets describing activities that happened at a specific location, or tweets introducing opinion about happening events, or any other text *i.e., "Hello, friends!!"*. The third type usually is not considered for analysis, as it is not meaningful for monitoring, recommendation, and event detection systems.

Tweet location prediction problem gained interest of researchers over last years. Researchers working on such field have tried different approaches to geotag tweets. Evolution of text mining techniques and natural language processing methods helped in improving text localization and geotagging. Many research initiatives have addressed the problem of location prediction from tweets. In [23] location prediction problem is categorized either as *Home location prediction* which refer to Twitter users' long-term residential addresses, or *Tweet location prediction* which means the place where a tweet is posted, or the *Mentioned Location prediction* as users may mention the names of some locations in tweet contents.

Some researchers have proposed convolutional deep neural networks [12] to learn Location Indicative Words (LIW) from word embedding. Others proposed a text network [11] consisting of Bidirectional LSTM to find out the geolocation of text. In addition, some proposed a Unicode network [8] with character encoding to find location of text in any language. While in [14] they have used kernel densities to estimate text location. Moreover, location prediction problem is either considered as a classification problem when the aim is to assign a label *i.e.,* a cell in grid map, or is considered as a

GeoAI '22, November 1, 2022, Seattle, WA, USA

ACM ISBN 978-1-4503-9532-8/22/11…$15.00
https://doi.org/10.1145/3557918.3565875

[1] https://developer.twitter.com/en/docs/twitter-api

regression problem when we want to get the geolocation coordinates. Those research initiatives have tried to reduce the distance error between real and predicted locations. They have reported good results, but still their accuracies are reported at wide ranges such as (5 KM, 10 KM, 20 KM, etc.).

Our objective is to **find the location where a tweet was posted**, thus our work can be categorized as *tweet location prediction at a finer spatial granularity level than the state-of-the-art.* Indeed, predicting the country or the city of a text becomes easier nowadays with the variety of proposed methods, while identifying the location at finer granularities is still a hard task. Therefore, we are motivated in **predicting tweet location restricted to a specific region**, e.g., a given city or an island. In other words, the city label of the tweet is known and the aim is to identify the location of this tweet within this city. Many of geotagged tweets provided by twitter API don't contain the precise location, they are geotagged within a coarse polygon which represents the city boundaries. Moreover, getting such information (i.e. city label of a tweet) is no more a hard task, as the existing geotagging approaches can identify the city label of a tweet accurately [7, 8], but they fail to geolocate tweets/text more precisely.

Several studies have used spatial models either by finding the landmarks to geotag the tweets or learning some location indicative words (LIW) such as in [3, 12, 14, 19]. On the other hand some researchers have used textual models to derive the location such as in [7–9, 13]. In our work, we tried to combine both assumptions to increase the accuracy and better geotag the tweets.

We propose *FLAIR* a "Fine-grained LocAtIon pRediction" algorithm of non geotagged tweets based on multi-view learning. Precisely, given a grid map, we predict the grid cell where a non-geotagged tweet was posted, thus reducing our location prediction problem to a classification problem. This classification method also **considers the location related text**, by using different natural language processing tools. We want to make use of the tweets in event detection systems, thus **our goal is to reduce the distance error to the actual location as much as possible**.

FLAIR is built on top of two factors. The first one is based on the assumption that a user talking about a certain location is quite likely to mention it in a tweet. The second factor taken into account is that tweets originating from the same location may be very relevant to one another because they may be discussing the same event. Based on those aspects, our work considers two models: a spatial model and a textual model. For the spatial model, we have used two methods either matching the POIs with the location entities of the tweets, or learning the cell indicative words in the map. While for the textual model, we compute the text similarity of non-geotagged tweets with other geotagged ones to identify the location. Finally, a multi-view model is adopted to combine their results and maximize the accuracy.

Our solution, adopts the multi-view learning approach as we consider two views to learn from. Each tweet is modeled in two different ways. We extract all possible entities in tweet text to be validated by the spatial classifier. Moreover, we calculate the text embedding of each tweet to compare its similarity with others. Then we use the stacking generalization approach in order to combine the results and have the final prediction.

FLAIR has been implemented and validated on real life tweets data that were collected within the context of GOGREEN ROUTES[2] project. The goal of this H2020 project is to provide guidance to cities in identifying and developing nature-based solutions to urbanization challenges by fostering urban mental health and well-being. Modeling people activities and events in different areas is a key to understand the urbanization challenges. This task can be easier with the existence of geotagged social media (GTSM) data.

The paper contributions can be summarized as follows:

- We propose *FLAIR* a novel algorithm to achieve a fine-grain tweet location prediction within a pre-defined region.
- We combine two prediction models by adopting multi-view learning: the one based on POIs matching (spatial model) and the other using text similarity (textual model).
- We further optimise the spatial model relying only on the geotagged tweets, and show their importance in the prediction process.
- We validate *FLAIR* on two real life datasets collected from Twitter, and compare the results with baseline and with the most similar approaches in the state-of-the-art, which clearly shows the advantages of *FLAIR* in terms of accuracy and spatial resolution.

The rest of this paper is organized as follows section 2 states the different methods of identifying tweet location in the literature, section 3 presents an example of our proposed solution and formally defines our approach. In section 4 we detail the different models used in *FLAIR*. Section 5 shows the implementation part and the experimental study. In 6 we discuss the shortcomings of the approach and the potential improvement. The conclusion and future work are drawn in section 7.

2 RELATED WORK

Considering the high importance insights one can extract from geotagged social media data, we are witnessing an increasing interest from both academic and industrial parties to the problem of tweet location prediction. Many existing strategies have been investigated in the study on location prediction from tweet and social media content, and we will discuss some.

In [23] have introduced an overall picture of the different families of location predictions performed on twitter data. In their work, they have illustrated the different types of inference that can take place depending on the tweet content, the twitter network, and the tweet context and meta-data. They presented the approaches proposed for predicting user home location depending on his/her previous tweets and friends, tweet location the location where a tweet was posted, and mention location prediction. Moreover, they differentiate between the evaluations metrics that can be used in location prediction problem, and grouped them either into distance-based metrics or token-based ones.

In [9] the authors proposed an approach relying on Language Model (LM) that is built by calculating term occurrence probabilities from processing a massive amount of geotagged items of a training set. The initial LM is further refined through feature selection and weighting. Terms are ranked and filtered based on accuracy, spatial

[2] https://gogreenroutes.eu/

entropy and locality. Then location estimation system employs two more steps (multiple grids, similarity search) oriented to achieve accurate location estimation. This model have shown good results, but it is dependent on the language model trained on. Authors have trained the language model on Flickr Images, and when they applied it to tweets, it did not show the same results. This approach is dependent of the training dataset, and it cannot be applied to any piece of text.

Chi et al. [3] proposed an algorithm to predict the location of Twitter users and tweets using a multinomial Naive Bayes classifier trained on Location Indicative Words and various textual features (such as city/country names, #hashtags and mentions). Authors have tried different combinations of features to find the best combination. Their reported results show that they still have a high distance error, while monitoring and event detection applications requires more precise location prediction.

In [12] authors proposed a Convolutional Neural Network (CNN) architecture for geotagging tweets to landmarks, based on the text in tweets and other meta information, such as posting time and source. They proposed an algorithm for geotagging tweets to landmarks. This algorithm requires tweet text represented as word embedding, source, creation time, and user location. The CNN network maps the input into a set of pre-defined list of POIs. This approach reports good results. This approach can work only when we have landmarks, thus if we do not have a landmark at some places we will not be able to geolocate the tweet.

An end-to-end neural network to predict the geolocation of a tweet was proposed in [11]. Their model is language independent, and requires six features the tweet text and other meta-data information. The proposed network has the capacity to automatically learn location indicative words and activity patterns from different regions. They used a character-level recurrent convolutional network for the tweet message. On the other side, in [19] the authors have followed the same architecture of the proposed model but replaced the character-level recurrent network with the *Word2Vec* embedding. Although **DeepGeo2** has increased the accuracy of location predictions, but for monitoring applications we still need predictions at a higher accuracy.

In [13] the author proposed a BiLSTM regression model neural regression model that can identify the linguistic intricacies of a tweet to predict the location. To identify the location of a given tweet, a double regression approach is adopted to identify the latitude and longitude separately for a given text. They used TF-IDF weighting to focus on highly relevant tokens in the text, and use FastText model to calculate embedding. This work is considered as regression as the aim is to assign the latitude and longitude of the text, and the results reported show that we still have a noticeable distance error.

Ozdikis et al. proposed a kernel density based location prediction method for tweets based on the geographical probability distribution of their terms over a region in [14]. Probabilities are calculated using Kernel Density Estimation (KDE), thus the prediction of tweet's location is performed by combining the probability distributions of its terms. This approach shows an improvement in accuracy at 5 KM, but fails to predict with the same accuracy at finer granularities.

In EDGE [7], authors cast the geolocation problem as a neutral network optimization problem by learning probabilistic generative models. EDGE consists of three main parts entity embedding extraction, attention aggregation, and mixture distribution learning. The authors states that the inference is built upon mining the correlation between non geo-indicative entities and geo-indicative entities, then each prediction result is returned as a Gaussian mixture rather than specific geographical coordinates. EDGE reported better accuracy than the state-of-art approaches, but still this accuracy is at 3 and 5 KMs.

Izbicki et al. [8] proposed a method for geo-locating tweets in any language. UnicodeCNN generates features directly from the Unicode characters in the input text. After passing the input to the convolutional layers, the softmax layer predicts the tweet's country of origin, and Mixture of von Mises-Fisher distributions is used to predict the GPS coordinates. This approach reported a high accuracy in predicting the location at country level and even city levels, but it fails to predict with a high accuracy at finer granularities.

In [4] authors utilized millions of Twitter posts and end-users domain expertise to build a set of deep neural network models using natural language processing (NLP) techniques, that predicts the geolocation of non geo-tagged Tweet posts. Their contribution was to provide a novel text modeling approach informed with feedback to predict the geolocation information. Different levels of granularities are covered through this work, the authors tried to predict the location information at the neighborhood level, the zip code level, and they also tried to predict the precise location by predicting the longitude and latitude. The reported accuracy for predicting the coordinates was very low, however the accuracy improved at the neighborhood and zip code granularities. The measured accuracy is at 30 miles, which considered a wide range while the aim is to minimize the distance error between the real and predicted locations.

Gonzalez et al. [6, 15] proposed a majority voting method that compares the similarity between non-geotagged tweets and geotagged ones. Their location inference model is based on a ranking approach combined with a majority voting of tweets weighted based on the credibility of its source. They estimated the geographical location of a given non-geo-tagged tweet by collecting the geo-location votes of the geo-tagged tweets that are most similar regarding their contents to that tweet. This model has a high accuracy at fine-grained granularity, but it takes into account only the textual aspect. While our intuition is to enhance such models and combine both spatial and textual aspects to improve the models accuracy.

Different approaches and methods were proposed to tackle the problem of text and tweet location prediction. Those approaches mainly success in predicting the city and country level of tweets, while failing at predicting location at finer granularities with a high accuracy. Most of the work focuses on location indicative words and word embedding, while not taking into consideration the relation between non-geotagged tweets and other geotagged ones in the same spatial area. In order to benefit from tweets in monitoring and event detection systems, we should have their precise locations. Hence, we need a text location approach that minimize the distance error between real and predicted locations.

3 METHODOLOGY

This section presents the followed methodology of predicting tweets' location. First subsection provides an illustrative example explaining how our approach works. Then in the overview subsection we define our problem formally and present the followed algorithm.

3.1 Illustrative Example

Table 1 shows an example of tweets that can be posted. Figure 1 illustrates a case example of how the proposed solution will predict the location of tweets. Tweets in table 1 refers to figure 1, so tweet 1 in the figure is tweet number 1 in the table.

As shown in figure 1, the map is splitted into cells using the grid view. The POIs are identified and the geotagged tweets are grouped into clusters based on their text similarity. Now to predict the location of non-geotagged tweets, we apply steps 6 through 9 in algorithm 1. For the first tweet, we can identify a location entity in the tweet text which is *"KFC"* and the tweet is not relevant to any of the geotagged clusters, thus it will be assigned the label 1 as its cell prediction (here the POI "KFC" is found in cell number 1). The second tweet, is talking about pollution and at the same time, there is many geotagged tweets talking about the same topic at a specific place (cell number 6). Even though the tweet did not mention any POI in the text but depending on sentence similarity, we can assign it to cell number 6 in the grid (we are assuming that cell number 6 is the cell where we have a conference about pollution). For the third tweet, we have a mentioned POI, in addition we have a cluster of geotagged tweets (here the tweets are about a car accident) that match the same topic of the third tweet, then we use the multi-view learning approach to assign cell 13 as the prediction. The case of the forth tweet is typically the same as the second tweet. While for the fifth tweet in the table, we will not be able to predict the location. Here, we can assign the user's home location if exists. Usually tweets like tweet number five in the table are not informative for monitoring application or detecting local events and activities.

Tweet number	Text
1	Having lunch at KFC!
2	Pollution monitoring conference. #GOGREEN
3	Ohh.. Traffic jam again near the castle.
4	Car accident near our home...
5	I am happy today!

Table 1: Example of Tweets

3.2 Overview of *FLAIR*

In this subsection, we will introduce the followed methodology to predict location of tweets. Given a pre-defined and well-known region of interest, we are interested in minimizing the distance error between the anticipated and the real locations of the tweets, as the aim is to utilize such predictions for event and activity detection. Predicting tweet location at country or city level is now possible, while predicting a precise location within a specified region is still a hard task. We address this challenge and evaluate the trade-off between the prediction accuracy and the spatial resolution.

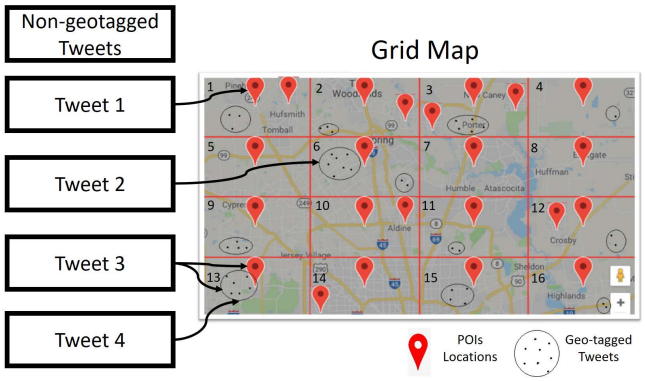

Figure 1: Tweet Location Prediction

A tweet contains mainly the tweet's text, a link to media if exists, mentions and hashtags, and a set of metadata. Metadata attached to the tweet are informative data, such as, id, creation time of the post, geo-graphical coordinates (if exists), language, likes, users id, etc.

We consider a tweet T defined by its text, creation time, and location T:<X,C,L> where X denotes the text, C denotes the creation time, and L denotes location. The overal objective is to predict L for the tweets while minimizing as much as possible the distance error between L and their real location.

Following the fact that twitter's users usually mentions point of interest in their tweets, we will detect the mentioned locations and match them with POIs found in the region. Moreover, usually tweets originating from the same place are highly probable to be relevant to each other. For instance, if there is a football match most of the tweets in the region of stadium will be talking about the match. Depending on those facts, we are proposing a model that takes into consideration POIs mentioned in tweet text on one hand, and on the other hand another model that finds tweet text similarity with other geotagged tweets. We consider each classifier here as an independent view, thus we adopt the multi-view learner to estimate the final location of a tweet.

Algorithm 1 states the steps of *FLAIR* to predict the location of a tweet. First, we need to identify an area of interest so we can split its map into cells given a granularity (Line 1). Those cells are considered our labels in the classification problem. We find all the points-of-interest (POIs) in this region, so we can assign them to their corresponding cells in the map (Line 2). Next, we train a spatial model using the POIs collected data (Line 3). Also, we collect all the geotagged tweets within the specified region, and assign labels to them based on their locations (Line 4). Then, we merge tweets within each cell based on their similarity, so we can form clusters of events (Line 5). At this phase, we exclude global events to keep only local ones (Line 6). After that, for each non-geotagged tweet, we compute its text similarity with the events' clusters formed in the previous step. The label of the most similar cluster will be assigned as the tweet label (Line 7). At the same time,

we check if there is POIs mentioned in the tweet text. Those POIs are matched to the POIs found in the map cells, and a second label is attached to the tweet (Line 8). In case a tweet neither matches any geo-located tweets cluster nor matches a POI in the grid cells, we assign it the user's location if exists (Line 9). Finally, we use a stack generalization approach that considers both views the spatial view (*"POIs Matching"*) and textual view (*"Sentence similarity"*) and uses their predictions to predict the final grid cell label (Line 10).

Algorithm 1: Methodology to predict tweets' location

Input: X, C, *granularity*
Output: L

1 Split the map of pre-defined region of interest into cells using a grid view, based on the given granularity.
2 Find the PoIs in each grid cell, and assign them cell labels.
3 Train a spatial model based on POIs to predict cell number.
4 Collect all geotagged tweets in the region, and assign them the appropriate cells labels.
5 Merge similar tweets in the same cell to form events clusters, depending on their creation time C and their text X.
6 Rank the clusters to identify true local events.
7 Calculate text similarity between non-geotagged tweets and identified clusters.
8 Match POIs mentioned in tweets with POIs found on the cells of the grid map.
9 If tweet's text doesn't match the 2 criteria above assign user home location if exists.
10 Use multi-view learning approach to predict final cell number (L).

4 THE MULTI-VIEW LEARNING STEPS

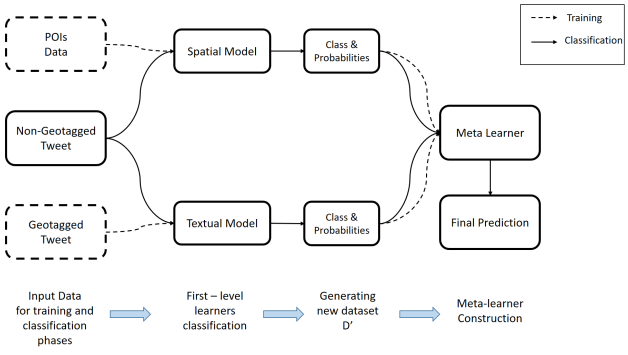

Figure 2: Multi-view Learning Approach

This section, details our tweet location prediction approach based on multi-view learning approach and using different natural language processing tools. Figure 2 shows different components of our location prediction work. Our tweet location prediction mainly depends on text, in addition we only use creation time of tweet from the metadata. We consider 2 views: the spatial model considered as the first view, the textual model is considered as the second view, and the multi-view model combines the results of the previous views. The dashed arrows represents the training phase of our multi-view learning model while the lines corresponds to the classification phase. *POIs data* and *Geotagged* tweets are used to train the spatial model and textual model respectively. Spatial model and textual model are considered as the first level learners, in their turn those models will predict the location of the tweet along with the probability. The predictions and probabilities are merged together to representing the feature vector as shown in table 3 in the new dataset D'. Finally a meta-learner is trained on top of D' to give the final prediction. For classification *Non-Geotagged tweets* are the input data for the model, they will be validated by all first level classifiers. The predictions and probabilities will be the input of the meta-learner, which will predict the location of the tweet.

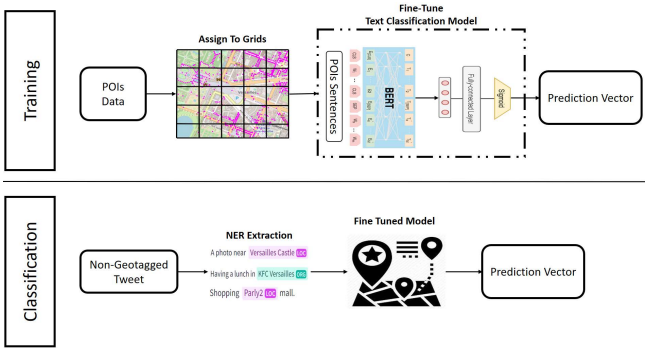

Figure 3: Spatial Model

For both types of collected data, twitter data and POIs data, we assign them labels. For this process, we have splitted the map (*i.e.* the area-of-interest or the bounding box) into cells using a grid view. A number identifies each cell in the grid view, and this number will be the representative label. Using the latitude and longitude geographical coordinates present in POIs and twitter data, we will match each record by its correspondent cell and we will assign the cell number to each record as its label.

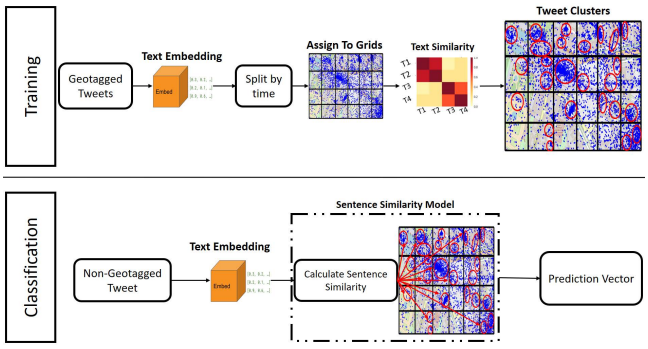

Figure 4: Textual Model

Figure 3 zoom in the details of the spatial model, we have a training phase demonstrating the steps performed to train this model and the classification phase shows how this model is used. First of all, this model is trained using POIs collected data. We form up

name	lat	lon	amenity	highway	place	city	street	landuse	shop	operator	Public Transport	country
CIC	48.796	2.136	bank	NaN	NaN	Versailles	Rue des Chantiers	NaN	NaN	CIC	NaN	France
Pharmacie Porchefontaine	48.796	2.154	pharmacy	NaN	NaN	Versailles	Rue Coste	NaN	NaN	NaN	NaN	France
Versailles Montreuil	48.803	2.153	post_office	NaN	NaN	Versailles	Rue Champ Lagarde	NaN	NaN	La Poste	NaN	France

Table 2: POI Dataset

sentences from the POIs data (*i.e.* place name, road name, city, country…), to generate a dataset that will train the text classification model. The sentences are generated by concatenating the name of the POI and the name of the street, the city, and the country with a space separator. Table 2 presents the different attributes found in POI dataset. For example, for the *CIC Bank* POI which has the following attributes in Open Street Map (OSM): (name: CIC, amenity: bank, highway: NaN, place: NaN, street: "Rue des Chantiers", landuse: NaN, operator: CIC,...., city: Versailles, country: France), we will form the following sentence: "CIC Rue des Chantiers Versailles France". The process of training the model is independent from the tweets. We have fine-tuned a text classification model to predict the cell number in a map. Fine-tuning is a known task in the world of NLP, it is tuning the model to predict outputs depending on a given dataset. We have used an already pre-trained BERT model distilbert-base-uncased[3][17], which was trained on a huge dataset. Dataset is tokenized to be used by the model for training. Tokenizer will tokenize the inputs by converting tokens to the corresponding ids in the model vocabulary, and it generates some other inputs required by the model (*i.e. attention mask*). The labels of those inputs are the corresponding cell numbers in the grid map.

For the classification part, for each tweet NER (Named Enitity Recognition) is applied to extract relevant location and organization entities. We will generate sentences out of the recognized entities, following the same process followed with POIs. The generated sentences are the input data for the fine-tuned model. The fine tuned spatial model will predict the location of the tweet based on the recognized entities present in the tweet, and its output is a prediction vector consisting of the prediction and the corresponding probabilities of each class. As this model is not trained on tweet's data, thus it can be used to predict the location of any text. At the same time training the model using POIs data can raise some problems caused by the dataset of twitter, since usually we may find errors and mistakes in the tweet's text. In addition, some can have a dialect and special way of tweeting. Those aspects can drop down the accuracy of the tuned model.

Figure 4 describes the training and classification phases of the textual model. The idea behind the textual model (sentence similarity approach) is to make use of existing geotagged tweets. We based our work following the fact that it is highly probable to have tweets discussing the same topic originating from the same place. Thus, we are trying to identify the location of non-geotagged tweets based on their topics similarity with geotagged tweets.

Discussed topic should not be a global event, because global events can be discussed anywhere. We need to identify the global events as a first step. To identify those events we can check the trending topics on twitter at a specific day and at specific country. Then all tweets related to those trending topics will not contribute in the work of this model. As shown in the training phase of figure 4 we split the geotagged tweets into groups depending on their creation time, and then each geotagged tweet will be assigned to its proper cell in the grid map based on its coordinates. Beforehand text embedding is calculated, and each tweet will be represented as an embedding vector. Text similarity among the tweets is calculated so we can form a groups of relevant tweets (i.e. discussing the same topic) at each cell of the grid. Classification phase describes how this model will be used, for each non-geotagged tweet we compute its similarity with geotagged ones taking place at the same time. Finally, the tweet will be assigned to the grid cell that maximizes its similarity with the geotagged tweets. The output of this model again is a prediction vector consisting of a predicted class and the vector of class probabilities.

The predictions of both models spatial and textual model will be a vector. In both figures 3, 4 the output of the model is the prediction vector. The prediction vectors after that are concatenated to generate a new dataset D' as shown in table 3.

As our aim is to enhance the prediction's accuracy and make use of the two methods proposed to find the location, we need to combine the results of both models. We consider each model as an independent view for predicting the location, given a specific input. The aim is to adopt a multi-view leaning approach in order to combine the results and maximize the accuracy. We adopt a multi-view learning approach inspired by stacking generalization approach [1, 5]. The idea here is to generate a new dataset based on the predictions of both models and their probabilities, and train a meta-model on top of them.

Table 3 shows an abstraction of the feature vector of the new dataset D'. The feature vector will contain the prediction of the different views denoted in the table as the first level learners, along with the probability of each learner (*i.e.* in other words we can describe it as the weight of this prediction), and the true label will be the actual cell number of each tweet. Random Forest classifier is used on top of this new dataset to train the meta-learner.

[3]https://huggingface.co/distilbert-base-uncased

Table 3: An example of the new generated dataset D'.

First-Level Learners						Prediction Probabilities						True Label
l_1	l_2	...	l_i	...	l_n	p_1	p_2	...	p_i	...	p_n	y

5 IMPLEMENTATION AND EXPERIMENTS

5.1 Implementation

This part details the implementation pipeline of our location prediction approach. The implementation includes four parts: data collection, data pre-processing, data enrichment, and model training and validation.

5.1.1 Data Collection. We have collected two types of real-life data within the context of GOGREEN ROUTES project. In this work we are interested in the tweets originating from two places *"Versailles"* and *"Santorini"*. Hence, we defined a bounding box for each region. We collect tweets originating from this defined bounding box, and collect all the POIs found inside this area.

For twitter data, we have collected tweets using the Twitter API. Using a research twitter account that allows collecting historical tweets and includes many other options.

To collect POIs inside the pre-defined bounding box we used the Open Street Map (OSM) data. Using Overpass [4] python library we have collected the POIs in the area-of-interest. This library will return the name, road, village, municipality, city, country, and latitude and longitude of each POI.

5.1.2 Data Pre-processing. For the tweets, we have changed the raw collected data into more meaningful views. We have exploited different tweets and transformed them to be in the form of attributes and values for each tweet. Then we applied link removal and text cleansing for all the tweets. Those links are the URLs of associated media with the tweets, usually they are attached to tweet text when crawling twitter data. Keeping those links may affect calculating the embedding of words and finding the sentence similarity. Moreover, we removed special characters and performed cleansing to the tweet text.

For the POIs collected data, we transform them into sentences using different combinations. Beforehand, we have identified the relevant POIs to keep and those to remove. To do that we relied on the type of the POI, so we removed some POIs such as traffic lights, traffic signs, etc. while keeping the relevant POIs (Restaurants, parks, stores, etc.). We have generated sentences from those POIs to train the spatial model. There is no standard way of how people mentions the POIs in their tweets. It is more commonly that they use only the name of the POI, or they use the POI name combined with city name or country name. For those reasons, we have generated most of the probable combinations that the user may write.

5.1.3 Data Enrichment. Enrichment is the process of adding knowledge to the dataset and transforming data into semantical views. In our case, we are dealing with qualitative data (text), thus we should extract the proper knowledge from the tweets. We enrich our data using two different ways, using NER extraction and embedding calculation.

To extract POIs mentioned in the tweet text we need the help of NLP techniques, thus we used the NER model. Specifically, we used the model developed for *NER* (Named Entity Recognition) that is found on huggingface [5] [6], which has around 88.5% overall precision. This model allows us to recognize the entities mentioned in the tweet text, such as "Persons, Organizations, Locations, etc…". Using organization and location recognized entities we will form a new sentences . Those sentences will be validated later by the spatial model to predict the label.

On the other side, to compute the textual similarity we should transform the raw text into embedding vectors. To perform this we have used a semantic model [16] found on huggingface [7], this model has an accuracy of around 87.4%. In this phase, we use the pre-processed text (text after cleansing and link removal). Each tweet will be represented as an embedding feature vector.

5.1.4 Model Training and Validation. This subsection describes the work done to train and predict the location of the tweet depending only on its text. We have three models to describe, a Spatial model will be used for POIs, textaul model is adopted to assign non-geotagged tweets to cells containing geotagged tweets, and a multi-view model is implemented to combine the results.

In data enrichment subsection 5.1.3, we have calculated the embedding of the pre-processed text and we assign each tweet a label (i.e. the cell number in the map depending on its real geo-coordinates). Also, we extracted all locations and organizations from the tweet text by the help of NER model. The extracted entities will form sentences and will be classified by the spatial model. While the embedding representation of each tweet will be used by the textual model.

Spatial model is a fine-tuned pre-trained BERT model trained using sentences generated from POIs data. It is a text classification model that takes a text as an input and outputs its corresponding cell in the map. On the other side, textual model is implemented to find the similarity of each non-geotagged tweet at a specific time with geotagged tweets at that time. Geotagged tweets are grouped into clusters, and we will calculate the cosine similarity of each non-geotagged tweet with all possible geotagged ones.

A random forest classifier is used as a meta-learner in the multi-view learning stage. We generate a new dataset based on the predictions of each view and the probability of each prediction to form the new feature vector. Random Forest classifier is trained on top of the new generated dataset to predict the final cell number.

5.2 Experiments

This part details the experiments performed and compares the accuracy of our work and the existing work. The models are implemented in Python 3.8 using Keras, Tensorflow, scikit-learn. All the experiments are carried on the same environment.

5.2.1 Experimental Settings. We evaluate the proposed model on real life data, we validated our approach using two datasets. We have collected tweets in the region of Versailles from 2010 until July 2021, and we have found around 370K geotagged tweets. Also,

[4] https://python-overpy.readthedocs.io/en/latest/

[5] https://huggingface.co/Jean-Baptiste/camembert-ner

[6] https://huggingface.co/xlm-roberta-large-finetuned-conll03-english

[7] https://huggingface.co/sentence-transformers/paraphrase-xlm-r-multilingual-v1

we have collected tweets in the region of Santorini between 2015 and 2021, and we have found around 186K geotagged tweets. Same experimental settings were used for both datasets, each dataset is splitted into 70% training set and the rest 30% are used for validation. For the area of interest (*i.e.* Versailles region, Santorini region), we have tried different grid splits. We started from 4 columns by 4 rows to split the map and reached 15 columns by 15 rows, then we performed a split using 20 columns by 20 rows, 30 by 30, 40 by 40, and reaching 50 by 50.

Table 4 reports some results of the performed experiments in Versailles region. The first 2 columns in table 4 shows the area of each cell and the number of classes found with respect to different granularities. Only cells containing tweets are considered, in fine granularity the number of labels increases while the area of cell decreases. For example at granularity (4x4) we have only 4 classes and the area of each cell is 21 KM^2, while for granularity (50x50) we have 227 different labels with an area 0.07 KM^2 for each cell. For the POIs, we have collected all the POIs in the area-of-interest and we kept only the relevant ones such as touristic places, restaurants, street names, etc.…

Each POI record has its corresponding geo-coordinates, using those coordinates we distribute the POIs over the cells of the grid map. For the geotagged tweets, the same approach is followed, tweets are distributed over the cells based on their geographical coordinates. Hence, in the experiments POIs and tweets are represented by the cell numbers (labels) and no more their real geographical coordinates.

5.2.2 Experimental Results. We consider the location prediction problem as a classification problem, where we aim to predict the cell label of the tweet instead of getting the exact geolocation coordinates. We evaluated our model against baselines and proposed approaches in the state-of-art for location prediction methods.

For the baselines we used the Multinomial Naïve Bayes Classifier (MNB) and Multilayer perceptron network (MLP) those 2 classifiers are considered as state-of-art methods for text classification, and as a baselines in location predictions problems. We validate our approach against other existing approaches in the literature. We considered two approaches Geotagging Tweets to Landmarks [12] and DeepGeo2 proposed in [11, 19]. We implemented both approaches using deep learning techniques, and we have reported the results on our dataset. For [12] we considered the labels (cell numbers) as the landmarks we want to predict.

Table 4 reports the experimental results of **Versailles region** . The spatial model trained on the POIs dataset has reported a high accuracy for all granularities when validate on POIs dataset, it reached around 82%. While when validating the model on locations extracted from tweets, the model accuracy drops down especially at finer granularities. For example, at the (4x4) granularity the accuracy was around 63.5%, at (10x10) it decreases to 40%, and reaches 5% at the (50x50) granularity. This decrease in the accuracy is due to two factors, the first is that the model have never seen the tweets data (as it is not trained on tweets data), and due to the way of writing on twitter as most users uses shortcuts to mention visited locations and usually they don't mention the specific location.

Textual model reported a good accuracy among different granularities. At all granularities it outperforms the baselines and existing approaches.

Combination of the results using the multi-view learning approach has enhanced the accuracy. As reported in table 4 our proposed approach has outperformed all the other classifiers at different granularities. It reports an accuracy of 71.6% at (50x50) granularity where the area is 0.07 KM^2.

This part reports the experimental results of **Santorini region** reported in table 5. Although our approach didn't shows the same accuracy in Santorini as in Versailles, but it still outperforms other approaches. Different granularities were tested starting from (5x5) and reaching (50x50), our approach shows an acceptable accuracy.

6 DISCUSSION

In this section, we discuss what are the shortcomings behind the decrease in accuracy for Santorini dataset and what are the perspectives for improving our approach.

The reported results shows the superiority of our approach when compared to others. For the two used datasets the multi-view model outperformed the existing approaches. For Versailles the results of our approach were better from those of Santorini. Although our approach has the highest accuracy for the dataset of Santorini, but the results weren't what we expected.

Santorini is a touristic place and most of the tweets in that region corresponds to tourists. We expected that the Spatial Model (i.e. the model trained on top of POIs) would have a better accuracy, since most of the tweets contain location entities.

There are two main reasons behind the low accuracy reported for the spatial model. First of all, the model is trained using POIs data, and validated on tweets data. The fine tuned model when validated on POIs data reports a high accuracy around 82%, but when validated on tweets data this accuracy drops down. The way of mentioning location in tweets doesn't look the same as the POIs data, this can explain the decrease in accuracy for the spatial model. Moreover, the second reason behind low accuracy reported is the language of the POIs dataset. In Santorini region we collected POIs data using the OSM API, but the retrieved data was in Greek, while the POIs mentioned in the tweets are mainly in English. Those aspects decrease the accuracy of the spatial model, thus the overall accuracy of the multi-view model drops down.

6.1 Spatial Model Adjustment

To cope with the low accuracy problem we proposed an adjustment for the spatial model. As mentioned the main problem is the difference between the form of POIs data and the way of mentioning those data in tweets. To solve this problem we proposed learning the location words of each cell from the historical geotagged tweets. Figure 5 shows the adjustment performed on our spatial model. In the training phase instead of using data collected from POIs we will perform NER on top of historical geotagged tweets so we can extract the location and organization entities in each tweet. After assigning the entities to the grid map, the entities will be identified by the labels. Then as done previously we will fine tune a text classification model to predict the cell number. For classification phase, we keep the same procedure as in the previous approach.

Granularity	Area KM^2	Number of labels	MNB	MLP	Geotagging to landmarks	DeepGeo2	Spatial Model	Textual Model	Multi View
(4x4)	21	4	69.8	66.1	71.8	78.4	63.5	83.4	**90.7**
(10x10)	2.3	25	39.3	47.8	45.7	62.6	39.5	68	**79.1**
(15x15)	0.9	46	35.1	47.1	43.8	54.6	33.2	65	**77.1**
(30x30)	0.2	125	29	45.6	40.9	51.9	6.9	62.9	**71.1**
(50x50)	0.07	227	21.3	34.3	42.3	36.4	5.1	59.6	**71.6**

Table 4: Accuracy of different models Versailles Region

Granularity	Area KM^2	Number of labels	MNB	MLP	Geotagging to landmarks	DeepGeo2	Spatial Model	Textual Model	Multi View
(5x5)	19.15	14	57	49	60.8	60	54.9	58.4	**68.9**
(10x10)	3.7	43	42.4	34.1	48.4	45.8	35.9	49.3	**56.1**
(15x15)	1.56	80	36.2	30.1	46.2	42.1	27.2	45.5	**54**
(30x30)	0.36	151	31.5	28	43.1	40.1	20.4	43.2	**52.7**
(50x50)	0.12	219	27.2	16.1	42	32.2	17.3	41.7	**50.5**

Table 5: Accuracy of different models Santorini Region

The spatial model now can learn the entities that are usually mentioned in different cells, in other words the model will learn the cell indicative words.

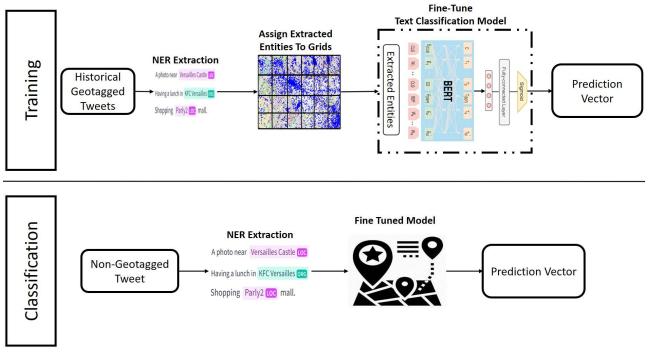

Figure 5: Adjusted Spatial Model

6.2 Results

To validate our work we repeated the experiments using the adjusted spatial model. Table 6 and table 7 reports the results of spatial model and multi-view model when using the first approach and the second approach for Versailles and Santorini respectively.

Reported results shows a significant improvement on the results of spatial model. For **Versailles region**, it is clear that the new approach improves the accuracy at finer granularity. Using spatial model (first approach) we had an accuracy of 6.9% and 5.1% for granularities (30x30) and (50x50) respectively, while with spatial model (second approach) we have an accuracy of 52.5% and 44.1%. The results improvement of spatial model is reflected on the results of multi-view model, thus we had a better overall accuracy at the different granularities.

Using spatial model (second approach) in **Santorini region** significantly improves the accuracy of the classification. The accuracy of the multi-view model (second approach) showed a good results when compared with the results of the first approach and the state -of-art methods.

Granularity	Spatial Model (First approach)	Spatial Model (Second approach)	Multi View (First approach)	Multi View (Second approach)
(4x4)	63.5	79.8	90.7	**93**
(10x10)	39.5	59	79.1	**84.3**
(15x15)	33.2	59.7	77.1	**83.3**
(30x30)	6.9	52.5	71.1	**78.9**
(50x50)	5.1	44.1	71.6	**78.5**

Table 6: Accuracy of different models Versailles Region

Granularity	Spatial Model (First approach)	Spatial Model (Second approach)	Multi View (First approach)	Multi View (Second approach)
(5x5)	54.9	76.9	68.9	**82.9**
(10x10)	35.9	66	56.1	**77**
(15x15)	27.2	61	54	**75.6**
(30x30)	20.4	53	52.7	**72.8**
(50x50)	17.3	52	50.5	**71.2**

Table 7: Accuracy of different models Santorini Region

7 CONCLUSION

Tweet location prediction has gained the interest of many researchers, especially for applications that uses social media data in analysis. Existing approaches succeeded to predict the location at city or country levels. In this paper, we have proposed *FLAIR* a multi-view learning approach for fine-grain tweet location prediction within a specific area of interest. Our approach is based on top of two models: spatial model which learns the location words from a tweet to find its location (either using POIs data, or extracted locations from historical tweets), while the textual model assign labels depending on text similarity.

Our approach requires minimal features, as it depends mainly on the tweet text. This approach can be adopted to any text corpus and not only twitter data. The reported results have shown that our model outperforms the baselines and existing approaches for location prediction problem. Especially when adjusting the spatial model, we obtain a significant improvement in terms of accuracy. The accuracy of the spatial model and that of the textual model drops down as the granularity decreases, but the combination of the results using the multi-view model shows an acceptable results for all granularities.

We are looking forward to enhance our approach by adding (at the first level learners) a new views such as media data. Indeed, using stack generalization approach allows adding or removing learners easily. We believe having other views on the data will improve the model accuracy, yet this needs to be evaluated.

8 ACKNOWLEDGMENTS

This work has been supported by the H2020 EU GO GREEN ROUTES funded under the research and innovation program H2020- EU.3.5.2 grant agreement No 869764. It has been also supported by the MASTER project that has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie-Slodowska Curie grant agreement N. 777695.

REFERENCES

[1] Mohammad Abboud, Hafsa El Hafyani, Jingwei Zuo, Karine Zeitouni, and Yehia Taher. 2021. Micro-environment Recognition in the context of Environmental Crowdsensing. *Proceedings of the Workshops of the EDBT/ICDT 2021 Joint Conference* 2841 (2021).

[2] Jie Bao, Yu Zheng, and Mohamed Mokbel. 2012. Location-based and preference-aware recommendation using sparse geo-social networking data, In Proceedings of the 20th international conference on advances in geographic information systems. *GIS: Proceedings of the ACM International Symposium on Advances in Geographic Information Systems*, 199–208. https://doi.org/10.1145/2424321.2424348

[3] Lianhua Chi, Kwan Hui Lim, Nebula Alam, and Christopher Butler. 2016. Geolocation Prediction in Twitter Using Location Indicative Words and Textual Features. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*. The COLING 2016 Organizing Committee, Osaka, Japan.

[4] Florina Dutt and Subhajit Das. 2021. Fine-grained Geolocation Prediction of Tweets with Human Machine Collaboration. *arXiv preprint arXiv:2106.13411* (2021).

[5] Enrique Garcia-Ceja, Carlos E. Galván-Tejada, and Ramon Brena. 2018. Multi-view stacking for activity recognition with sound and accelerometer data. *Information Fusion* 40 (March 2018), 45–56. https://doi.org/10.1016/j.inffus.2017.06.004

[6] Jorge David Gonzalez Paule, Yashar Moshfeghi, Joemon M Jose, and Piyushimita Thakuriah. 2017. On fine-grained geolocalisation of tweets. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*. 313–316.

[7] Bo Hui, Haiquan Chen, Da Yan, and Wei-Shinn Ku. 2021. EDGE: Entity-Diffusion Gaussian Ensemble for Interpretable Tweet Geolocation Prediction. *2021 IEEE 37th International Conference on Data Engineering (ICDE)* (2021), 1092–1103.

[8] Mike Izbicki, Vagelis Papalexakis, and Vassilis Tsotras. 2019. Geolocating Tweets in Any Language at Any Location. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China) *(CIKM '19)*. Association for Computing Machinery, New York, NY, USA, 89–98. https://doi.org/10.1145/3357384.3357926

[9] Giorgos Kordopatis-Zilos, Symeon Papadopoulos, and Ioannis Kompatsiaris. 2017. Geotagging Text Content With Language Models and Feature Mining. *Proc. IEEE* PP (08 2017), 1–16. https://doi.org/10.1109/JPROC.2017.2688799

[10] Sangeeta Lal, Lipika Tiwari, Ravi Ranjan, Ayushi Verma, Neetu Sardana, and Rahul Mourya. 2020. Analysis and Classification of Crime Tweets. *Procedia Computer Science* 167 (2020), 1911–1919. https://doi.org/10.1016/j.procs.2020.03.211 International Conference on Computational Intelligence and Data Science.

[11] Jey Han Lau, Lianhua Chi, Khoi-Nguyen Tran, and Trevor Cohn. 2017. End-to-end Network for Twitter Geolocation Prediction and Hashing. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Asian Federation of Natural Language Processing, Taipei, Taiwan, 744–753. https://aclanthology.org/I17-1075

[12] Kwan Hui Lim, Shanika Karunasekera, Aaron Harwood, and Yasmeen George. 2019. Geotagging Tweets to Landmarks using Convolutional Neural Networks with Text and Posting Time. In *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion*. Association for Computing Machinery, New York, NY, United States. https://doi.org/10.1145/3308557.3308691

[13] Piyush Mishra. 2020. Geolocation of Tweets with a BiLSTM Regression Model. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*. International Committee on Computational Linguistics (ICCL), Barcelona, Spain (Online), 283–289. https://aclanthology.org/2020.vardial-1.27

[14] Ozer Ozdikis, Heri Ramampiaro, and Kjetil Nørvåg. 2018. Locality-Adapted Kernel Densities for Tweet Localization. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 1149–1152. https://doi.org/10.1145/3209978.3210109

[15] Jorge David Gonzalez Paule, Yeran Sun, and Yashar Moshfeghi. 2019. On fine-grained geolocalisation of tweets and real-time traffic incident detection. *Information Processing & Management* 56, 3 (2019), 1119–1132.

[16] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. http://arxiv.org/abs/1908.10084

[17] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2019).

[18] Luke S Sloan, Jeffrey Morgan, William Housley, Matthew Leighton Williams, Adam Edwards, Peter Burnap, and Omer Farooq Rana. 2013. Knowing the Tweeters: Deriving Sociologically Relevant Demographics from Twitter. *Sociological Research Online* 18 (2013), 74 – 84.

[19] Luke Snyder, Morteza Karimzadeh, Ray Chen, and David Ebert. 2019. City-level Geolocation of Tweets for Real-time Visual Analytics, In Proceedings of the 3rd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery. *GeoAI 2019: Proceedings of the 3rd ACM SIGSPATIAL International Workshop on AI for Geographic Knowledge Discovery*, 85–88. https://doi.org/10.1145/3356471.3365243

[20] Catherine M. Vera-Burgos and Donyale R. Griffin Padgett. 2020. Using Twitter for crisis communications in a natural disaster: Hurricane Harvey. *Heliyon* 6, 9 (2020), e04804. https://doi.org/10.1016/j.heliyon.2020.e04804

[21] Chao Zhang, Liyuan Liu, Dongming Lei, Quan Yuan, Honglei Zhuang, Timothy Hanratty, and Jiawei Han. 2017. TrioVecEvent: Embedding-Based Online Local Event Detection in Geo-Tagged Tweet Streams. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, United States, 595–604. https://doi.org/10.1145/3097983.3098027

[22] Chao Zhang, Guangyu Zhou, Quan Yuan, Honglei Zhuang, Yu Zheng, Lance Kaplan, Shaowen Wang, and Jiawei Han. 2016. GeoBurst: Real-Time Local Event Detection in Geo-Tagged Tweet Streams. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. Association for Computing Machinery, New York, NY, United States, 513–522. https://doi.org/10.1145/2911451.2911519

[23] Xin Zheng, Jialong Han, and Aixin Sun. 2018. A Survey of Location Prediction on Twitter. *IEEE Transactions on Knowledge and Data Engineering* 30, 9 (Sep. 2018), 1652–1671. https://doi.org/10.1109/TKDE.2018.2807840

[24] Lina Zhou, Dongsong Zhang, Christopher Yang, and Yu Wang. 2017. Harnessing social media for health information management. *Electronic Commerce Research and Applications* 27 (12 2017). https://doi.org/10.1016/j.elerap.2017.12.003