Classification of museum-related tweets in New York City

Introduction

The aim of the project is to explore a relationship between location and a use of language in microblog data, specifically, Twitter data. Twitter data is open and it is also possible to query it based on location. One of important tasks is to identify and predict tweets with "semantically meaningful locations" (Lee et al., 2014), or points-of-interest locations, such as, for example, museums in New York City. The work presented in this paper aims to answer two questions: to which degree a text of a given tweet is related to an actual physical location of that tweet (whether there is a distance dependency), and what features could be used to predict or classify a tweet as related to a museum (as a topic/location). Two classification experiments are performed using differently balanced data. Several classifiers' performance is compared based on two feature models (a set of word co-occurrences and a dictionary of museum names, abbreviations and tags).

1 Related Work

The geo-tagged Twitter data has been a subject of many research studies. The closest research study to the topic of this project has been performed by German researchers who compare points-of-interest and Twitter text around those points (in a ~250 meters buffer radius) in Germany and also look at a distance dependency of the language in relation to those locations (Hahmann et al., 2014). This research takes an example of a railway station as one of points-of-interest – in a similar way, in this project I focus on a museum as a point of interest in New York City and a 250 meter radius around it to pull the geo-tagged tweets from the Twitter API.

One of interesting aspects of this topic is that museum-related keywords may appear not necessarily in a strict geometric shape of a circle (radius buffer) but rather remind a polygon of a different shape. Some researchers call this a social point-of-interest (Vu et al., 2016) and this is one consideration that should be taken into account when looking to predict a type of activity and proximity to a location from microblog text data.

There are examples of other research works where the authors aim to predict location of tweets by summarizing Twitter network (network of a given user), tweet content, and tweet context as inputs for prediction (Zheng et al., 2017). Several research works analyze distance dependency of a social network and use the distance decay notion to predict location of social network users (Backstrom et al., 2010) and describe socio-spatial properties of these networks (Scellato et al., 2011). Another study develops a tool that enables researchers to examine spatial variation of Twitter language used in different locations (Ljubesic et al., 2016).

There is a number of studies dedicated to Twitter location prediction on a state or city level (Han et al., 2014; Chi et al., 2016; Amitay et al., 2004). The positional accuracy has varied from 100 miles (Cheng et al., 2010) and 479 km (Wing and Baldridge, 2011) to 30 km (Shulz et al., 2013).

The closest paper to the nature of this project is by German researchers who set up their experiment with specific points-of-interest locations in order to identify whether the text contained in tweets correlates with those locations (Hahmann et al., 2014). I follow the experimental setup of this paper and use museums as points-of-interest locations in New York City and tweet content around those points to try to predict whether a user is at a museum location based on a text input of the microblog data. The authors in the German experiment describe the results of three types of classification: manual classification, supervised machine classification using manually classified training data, and unsupervised machine classification using lexical training data. Their most prominent example of a point-of-interest location classes is a railway station. The feature vectors are created based on a unigram-model, where features are words with corresponding weights, or probabilities for the class. The authors use a geo-tagged Twitter corpus downloaded within the region of Germany to perform, first, a manual labeling of data (tweet text identified as related or not related to a class "railway station"). This manually annotated corpus is then used to perform a supervised machine classification based on Naïve Bayes (NB) and Maximum Entropy (ME) algorithms. The researchers also try an unsupervised machine classification using a newspaper corpus lexical training data (unigram features with weights of whether related or not related to a class "railway station").

In this paper two classification experiments are performed with differently balanced data: 8%-92% and 50%-50%. Classifiers include Naïve Bayes, Decision Tree, Maximum Entropy and support

vector machine (SVC) classifier. Methodology differs in terms of two feature models: one includes a list of unigram word co-occurrences and another one contains a dictionary of museum names, abbreviations and tags. The results are compared with a random baseline (in a case of 8%-92% balanced dataset it's 85% accuracy and in case of a 50%-50% balanced dataset it is 50% accuracy).

2 Data

Data is collected using Twitter API and Tweepy package for Python with a specified geographic parameter (radius) around museum locations in New York City. In order to check for a distance dependency factor, a control sample of tweets near park locations is collected.

For a museum classification task, in total 7,033 tweets are collected, after removing exact duplicates. 583 tweets are labeled as museum. For a second experiment with a balanced data, another corpus is created out of a larger dataset with a 50%-50% distribution and a total of 1,116 tweets. The lists of museum and park locations in New York City are collected through NYC Open Data portal. Word co-occurrences are collected through the Wortschatz corpora portal (© 2018 Abteilung Automatische Sprachverarbeitung, Universität Leipzig).

The data structure of training and testing sets is similar to a Reuters Corpus in NLTK (© 2017, NLTK Project), where one document represents a tokenized tweet content with a class label ("museum", "non-museum").

3 Tools and Methods

For downloading data the following tools are used: Twitter API and Tweepy package for Python. For classification task, Naïve Bayes, Decision Tree, Maximum Entropy and support vector machine (SVC) classifiers are used through NLTK (built-in functions and SklearnClassifier).

In terms of methodology, two approaches to feature selection are compared: a list of unigram synonym/semantic sets of word co-occurrences and a dictionary of museum names, abbreviations and tags. Performance of different classifiers mentioned above is compared with a random baseline performance. In a case of an unbalanced dataset (8%-92%), the random baseline is 85%. For the balanced dataset, the random baseline is 50%.

4 Experiment

First, a distance dependency factor is checked using "near-museum" and "near-park" corpora. By doing a simple string search of a word "museum" in both corpora (Table 1), we can see that there is a higher probability to encounter this word in a "near-museum" corpus than in a "near-park" corpus (in a "near-museum" corpus within 100 meter radius 10% tweets contain word "museum", while within 100 meter radius of park locations there are 0% tweets that contain word "museum"). We can also observe a distance decay phenomenon where while getting further from the location, the frequency of word "museum" drops in the "near-museum" corpus (from 10% in a 100 meter radius it drops to 5% in a 250 meter radius, 1.48% in a 500 meter radius and 1.2% in a 1 km radius). For the "near-park" corpus the probability of "museum" word increases as the radius increases (from 0% in a 100 meter radius to 0.97% in a 250 meter radius, 1.4% in a 500 meter radius and 0.7% in a 1 km radius). The data for these two corpora is downloaded at the same time of the day to control for a time factor.

Radius	"Near-museum"	"Near-park" (control sample)
100 meters	10%	0%
250 meters	5%	0.97%
500 meters	1.48%	1.4%
1 kilometer	1.2%	0.7%

Table 1: Comparison of percentage of tweets that contain word "museum" in "near-museum" and "near-park" corpora.

The evaluation results for the classification experiment are presented in the next section. In terms of experimental setup, the text data is preprocessed – while such symbols as "@" and "#" are preserved as they may contain a museum name, the URL links, numbers and '/' symbol are removed.

Two types of features sets are created. The first one (Model 1) contains a unigram synonym/semantic feature set based on co-occurrences of words "museum" and "gallery" on a Wortschatz corpora portal (© 2018 Abteilung Automatische Sprachverarbeitung, Universität Leipzig). A second feature set (Model 2) includes a list with New York City museum names, abbreviations and tags. Two subsets of this list are

created: Model 2a represents a unigram model, where museum names are tokenized (e.g. ["American","Academy","of","Arts","and","Letters"], and such stopwords as “of” and “and” are removed); Model 2b could be called an entity model, where museum names are preserved as one token (e.g. ["American Academy of Arts and Letters"]). The accuracy results of different classifiers are then compared: Naïve Bayes, Decision Tree, Maximum Entropy and support vector machine (SVC). Most informative features are extracted from Naïve Bayes and Maximum Entropy classifiers’ results.

In total, two classification experiments are performed using unbalanced (8%-92%) and balanced (50%-50%) datasets.

5 Evaluation

The classification experiment on the unbalanced dataset returns better results with an average classifier accuracy of 95%, comparing with an 85% accuracy for random baseline results (Table 2). An average accuracy of 48% is achieved for the balanced data which is slightly lower than a random baseline accuracy rate of 50%.

	Unbalanced	**Balanced**
Random baseline	85%	50%
Averaged classifier results	95%	48%

Table 2: Comparison of accuracy results with random baseline for unbalanced and balanced datasets.

The performance results of different classifiers do not differ drastically for an unbalanced dataset (Table 3). For a balanced dataset, Naïve Bayes classifier performs best with 5% above 50% random baseline results, followed by Decision Tree (3% above random baseline) and Maximum Entropy (2% above random baseline) (Table 4).

Classifier	**Model**		
	1	**2a**	**2b**
Random Baseline	85%		
Naïve Bayes	91%	90%	91%
Decision Tree	91%	90%	91%
Maximum Entropy	91%	90%	91%
Support vector machine	91%	90%	91%

Table 3: Accuracy results – a sample of one testing run for an unbalanced dataset.

Classifier	**Model**		
	1	**2a**	**2b**
Random Baseline	50%		
Naïve Bayes	47%	55%	50%
Decision Tree	47%	53%	50%
Maximum Entropy	46%	48%	52%
Support vector machine	45%	48%	48%

Table 4: Accuracy results – a sample of one testing run for a balanced dataset.

In terms of feature models, the unigram model of a list of museum names (Model 2a) performs slightly better than synonym/semantic word co-occurrences (Table 4). Among the most informative features for synonym/semantic sets are words "exhibition”, “collection”, “visit", "photo", “art”. For the unigram model of a dictionary, most informative features are words "Schomburg", "National", "Research", “Institute”, “Technology”. For an entity model of the dictionary, most informative features are those with a museum name and a “@” tag (such as "@studiomuseum", "@brooklynmuseum", etc.).

Discussion/Conclusion

The distance dependency experiment in this paper shows that there is indeed a higher probability of encountering museum-related topic near museum locations.

The classification experiment shows interesting results in terms of balanced/unbalanced data. The averaged performance of classifiers is better on an unbalanced data overcoming a random baseline, while the averaged performance results are equal or less than a random baseline for a balanced dataset. This might have to do with the fact that in some cases a “naturally” unbalanced data could do better than an artificially constructed balanced dataset.

In terms of further work, as suggested by a reviewer Dr. Filatova, for distance dependency, beyond a control sample of another location, another topic could be used, for example, with a hashtag that is completely unrelated to a museum topic. Another suggestion is to remove features with “@”, as they seem to be overly informative, and look for other features in the same tweets.

Suggestions from another work (Lee et al., 2014) also include filtering for "location-neutral" tweets to reduce noise in data, as well as filtering

for future or past tense, potentially through part-of-speech tagging.

Acknowledgements

Thank you to Dr. Chodorow for advice and help with the conceptualization of the project, suggestions for data collection and the experiment as part of an independent study. Thank you to Dr. Filatova for all the advice and help with conceptualization, classification methodologies and techniques, paper formatting and finalizing this experiment during the class of Text Mining. Thank you to Dr. Rozovskaya for advice and suggestions on using WordNet or similar semantic/synonym sets for a feature model and sharing useful techniques and model reporting during Methods in Computational Linguistics II class.

References

Stefan Hahmann, Ross Purves and Dirk Buerghardt. 2014. Twitter location (sometimes) matters: Exploring the relationship between georeferenced tweet content and nearby feature classes. J. Spatial Information Science 9 (2014): 1-36.

Dung D. Vu, Hien To, Won-Yong Shin and Cyrus Shahabi. 2016. GeoSocialBound: an efficient framework for estimating social POI boundaries using spatio-textual information. *GeoRich@SIGMOD.*

Xin Zheng, Jialong Han and Aixin Sun. 2017. A survey of location prediction on twitter. *Computer Science Repository*, arXiv:1705.03172 [cs.SI].

Nikola Ljubesic, Tanja Samardzic and Curdin Derungs. 2016. TweetGeo – a tool for collecting, processing and analysing geo-encoded linguistic data. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3412–3421, Osaka, Japan, December 11-17 2016.

Axel Schulz, Aristotelis Hadjakos, Heiko Paulheim, Johannes Nachtwey and Max Mühlhäuser. 2013. A multi-indicator approach for geolocalization of tweets. In *International AAAI Conference on Web and Social Media, North America 2013*. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/view/6063.

Benjamin Wing and Jason Baldridge. 2011. Simple supervised document geolocation with geodesic grids. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 955–964, Portland, Oregon, June 19-24, 2011.

Zhiyuan Cheng, James Caverlee and Kyumin Lee. 2010. You are where you tweet: a content-based approach to geo-locating twitter users. In *Proceedings of the 19th ACM international conference on Information and knowledge management (CIKM '10)*. ACM, New York, NY, USA. http://dx.doi.org/10.1145/1871437.1871535

Bo Han, Paul Cook and Timothy Baldwin. 2014. Text-based twitter user geolocation prediction. *Journal of Artificial Intelligence Research* 49 (2014) 451-500.

Lianhua Chi, Kwan Hui Lim, Nebula Alam and Christopher J. Butler. 2016. Geolocation prediction in twitter using location indicative words and textual features. In *Proceedings of the 2nd Workshop on Noisy User-generated Text*, pages 227–234, Osaka, Japan, December 11 2016.

Einat Amitay, Nadav Har'El, Ron Sivan and Aya Soffer. 2004. Web-a-where: geotagging web content. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '04).* ACM, New York, NY, USA, 273-280. http://dx.doi.org/10.1145/1008992.1009040.

Lars Backstrom, Eric Sun and Cameron Marlow. 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 (WWW '10).* ACM, New York, NY, USA. http://dx.doi.org/10.1145/1772690.1772698

Salvatore Scellato, Anastasios Noulas, Renaud Lambiotte and Cecilia Mascolo. 2011. Socio-spatial properties of online location-based social networks. In *International AAAI Conference on Web and Social Media* (2011).

Ashequl Qadir and Ellen Riloff. 2013. Bootstrapped Learning of Emotion Hashtags #hashtags4you. In *the 4th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA 2013).*

Alexander Gelbukh. Ed. 2004. Computational Linguistics and Intelligent Text Processing: Proceedings of 5th International Conference CICLing (Berlin and Heidelberg, 2004), vol. 2945 of Lecture Notes in ComputerScience, Springer.

Uwe Quasthoff, Matthias Richter, Christian Biemann. 2006. Corpus portal for search in monolingual corpora. In *Proc. 5th International Conference on Language Resources and Evaluation*, N.Calzolari, Ed., pp. 1799–1802.

Kisung Lee, Raghu K. Ganti, Mudhakar Srivatsa, Ling Liu. (2014). When Twitter meets Foursquare: Tweet Location Prediction using Foursquare. In

MOBIQUITOUS '14 Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, pages 198-207, London, United Kingdom — December 02 - 05, 2014.

Twitter API. *https://dev.twitter.com*

Python Software Foundation. Python Language Reference, version 3. Available at http://www.python.org

Tweepy package. Available at: http://www.tweepy.org/

Natural Language Toolkit. https://www.nltk.org/

NYC Open Data portal. *https://opendata.cityofnewyork.us*

stackoverflow.com/questions/27900451/convert-tweepy-status-object-into-json#comment44408594_27901076

stackoverflow.com/questions/42225364/getting-whole-user-timeline-of-a-twitter-user

stackoverflow.com/questions/43038187/python-tweepy-cursor-jsonparser-object-has-no-attribute-model-factory

github.com/tweepy/tweepy/issues/538

machinelearningmastery.com/dont-use-random-guessing-as-your-baseline-classifier

stackoverflow.com/questions/3094659/editing-elements-in-a-list-in-python

stackoverflow.com/questions/24399820/expression-to-remove-url-links-from-twitter-tweet/24399874

bytes.com/topic/python/answers/670554-how-print-raw-string-data-variable

stackoverflow.com/questions/1919044/is-there-a-better-way-to-iterate-over-two-lists-getting-one-element-from-each-l

stackoverflow.com/questions/6304808/how-to-pass-tuple-as-argument-in-python

stackoverflow.com/questions/19560044/how-to-concatenate-element-wise-two-lists-in-python

stackoverflow.com/questions/716477/join-list-of-lists-in-python

stats.stackexchange.com/questions/227088/when-should-i-balance-classes-in-a-training-data-set

datascience.stackexchange.com/questions/17910/bad-classification-performance-of-logistic-regression-on-imbalanced-data-in-test

win-vector.com/blog/2015/02/does-balancing-classes-improve-classifier-performance/