# Mining the Web for Points of Interest

Adam Rae
Yahoo! Research
adamrae@yahoo-inc.com

Vanessa Murdock
Yahoo! Research
vmurdock@yahoo-inc.com

Adrian Popescu *
CEA, LIST
adrian.popescu@cea.fr

Hugues Bouchard
Yahoo! Research
bouchard@yahoo-inc.com

## ABSTRACT

A *point of interest* (POI) is a focused geographic entity such as a landmark, a school, an historical building, or a business. Points of interest are the basis for most of the data supporting location-based applications. In this paper we propose to curate POIs from online sources by bootstrapping training data from Web snippets, seeded by POIs gathered from social media. This large corpus is used to train a sequential tagger to recognize mentions of POIs in text. Using Wikipedia data as the training data, we can identify POIs in free text with an accuracy that is 116% better than the state of the art POI identifier in terms of precision, and 50% better in terms of recall. We show that using Foursquare and Gowalla checkins as seeds to bootstrap training data from Web snippets, we can improve precision between 16% and 52%, and recall between 48% and 187% over the state-of-the-art. The name of a POI is not sufficient, as the POI must also be associated with a set of geographic coordinates. Our method increases the number of POIs that can be localized nearly three-fold, from 134 to 395 in a sample of 400, with a median localization accuracy of less than one kilometer.

## Keywords

geographic information extraction, mobile devices, local search, location-based applications, points of interest

## 1. INTRODUCTION

A *point of interest* (POI) is a focused geographic entity such as a landmark, a school, an historical building, or a business. Many POIs are permanent structures such as statues, or buildings. Others are semi-permanent, such as restaurants which may open one year and close the next. Still others are temporal or periodic, such as the location of an annual festival.

Points of interest are the basis for most of the data supporting location-based applications. They are displayed on online maps, and are provided to users of location-based mobile applications such as Foursquare[1] and Gowalla.[2] POIs are often displayed in the local search results or in sponsored listings related to a user's search, or are presented to users of recommender systems such travel booking websites.

In this paper we address the question of whether it is possible to automatically identify POIs without manual intervention, given their broad definition, and the lack of consistency in their naming. Further, as knowing the name of a POI is not useful without knowing its location, we address the question of how accurately POIs can be localized using automatic methods.

To curate POI data, companies such as Navteq[3] and TeleAtlas[4] send a surveyor to a location to identify, verify and record POIs. This process yields high-quality and accurate data, although the process is very expensive, and it is difficult to scale to large geographic areas, or hard-to-reach locations. Furthermore, the frequency with which the locations can be surveyed is limited by the expense of gathering the data, so the process focuses on POIs that are more likely to be permanent, in areas of high commercial value. This creates a bias in the data towards POIs such as landmarks, schools, hospitals and other stable, stationary items that are unlikely to change significantly within a short time frame. Other types of POIs, such as restaurants, may change more quickly than human geographers can be sent out to update their records, and hence would become stale. For this reason they are not typically covered in this type of curated data.

A second method for curating POIs is to create a directory of sponsored listings. Directories of this type are used in local search engines and mapping products such as those that display restaurants and businesses. This second type of data is dependent on businesses that have the budget and inclination to pay to be included in the directory. Very small, independent businesses may be less likely to be listed, and in countries where Internet usage is low, the sponsored listings may be non-existent or sparse, or dominated by large businesses and national chains. As a result, the representation

*Work performed while the author was a visitor at Yahoo! Research

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
*SIGIR* '12 Portland, OR
Copyright 20XX ACM X-XXXXX-XX-X/XX/XX ...$10.00.

[1] www.foursquare.com visited February 2012
[2] www.gowalla.com visited February 2012
[3] `http://www.navteq.com/` visited February 2012
[4] `http://www.teleatlas.com/` visited February 2012

of the place in terms of local businesses will be far from the user's perception in real life.

A third technique is crowd-sourcing, such as the OpenStreetMap Project[5], in which users themselves contribute to the representation of the place by uploading location data. This helps address the bias toward large businesses and permanent POIs, but crowd-sourcing introduces other issues such as the provenance of the data, and its reliability. For example, the New York Times reported that Google erroneously labeled businesses as "closed" on their maps, as a result of spamming by competing businesses [17].

In this paper we propose to curate POIs from online sources by bootstrapping training data from Web snippets, seeded by POIs gathered from social media. This large corpus is used to train a sequential tagger to recognize mentions of POIs in text. Using Wikipedia data as the training data[6], we can identify POIs in free text with an accuracy that is 120% better than the state of the art POI identifier in terms of precision, and 48% better in terms of recall. The points of interest represented in Wikipedia resemble curated data found in gazetteers, and focus on official names for permanent structures. Businesses, restaurants and venues are under-represented in Wikipedia, but are well-covered in online checkin services such as Foursquare and Gowalla. We show that using Foursquare and Gowalla checkins as seeds to bootstrap training data from Web snippets, we can improve precision between 16% and 52%, and recall between 48% and 187% over the state-of-the-art.

The name of a POI is not sufficient, as the POI must also be associated with a set of geographic coordinates. We localize the POI mentions using location models inferred from Flickr data. The result is a method for discovering POIs previously not found in sponsored listings or online gazetteers. Our method increases the number of POIs that can be localized nearly three-fold, from 134 to 395 in a sample of 400, with a median localization accuracy of less than one kilometer.

To summarize, our contributions are as follows:

- A method to identify mentions of POIs in online data that is significantly better than the state-of-the-art in terms of both recall and precision. Our method does not require editorial labeling of data, and allows for the discovery of temporary or ephemeral POIs, in addition to more permanent structures.

- A method to increase the amount of training data by bootstrapping from Web snippets, that is language independent, and can be targeted to a particular region of the world. This allows for vast amounts of training data to be collected, without the overhead of manual labeling.

- A method to localize POIs that is highly precise, with a recall of nearly three times that of the state-of-the-art. The method is language independent, and does not depend on a database of sponsored listings, or human surveyors.

We present an overview of the related work in Section 2, followed by a description of the model for identifying POIs in text in Section 3. While the models are trained on data that is automatically generated, the evaluation is conducted on data labeled by human assessors. Section 4 presents the creation of manually annotated evaluation data, as well as training data gathered from Wikipedia. We present the bootstrapping system in Section 5 to extract POI mentions from social media data, and use those as seeds to create training data from Web snippets. The localization component is presented in Section 6, and the work as a whole is discussed in Section 7. We lay out our conclusions and reflect on our findings in Section 8.

[5]www.openstreetmap.org visited February 2012

[6]www.wikipedia.org visited February 2012

## 2. RELATED WORK

To the best of our knowledge there is no other work that aims to discover points of interest from unstructured text on the web. This section discusses related work that leverages the location of a user and his interaction with a map to improve the metadata associated with a point of interest. Although the literature on Named Entity Recognition is certainly relevant here, we included the related work on NER in the section describing the models. The localization component is based on the work of Serdyukov et al. [18], but there are other similar works that are also relevant to the localization of place mentions.

Mummidi and Krumm discover points of interest from pushpins placed on maps by users [15], mining the annotations of the pushpins for terms with a high TF·IDF value. The authors propose the map data as a reliable source of data from users, because the users have explicitly indicated a point of interest on the map, and after processing the data yields a textual characterisation of the point, plus its geographic coordinates. Evaluating a data discovery system is always a difficult task and the authors address this by conducting a user study in which 100 users assess points of interest shown on a map in their neighbourhood and are asked to indicate whether the POI is identified correctly or not.

In related, more recent work, Zheng et al. [26] propose a method to mine GPS data to recommend locations to users wanting to do an activity and to recommend activities to users at a particular location. Their data is obtained from an interactive mapping application, with 162 users, who have generated roughly 12,000 trajectories in Beijing over the past 2.5 years. In their data, the points of interest from a database of POIs are associated with geographic coordinates.

Both of these works rely on users' interactions with a map and are based on a small-scale user study. This is a key difference between our work and theirs as they rely on users to annotate maps with POI data. Our work discovers the mentions of POIs in Web snippets and does not rely on users interacting with any particular application. This allows us to potentially gather vast amounts of training data, independent of any given application. Although the POIs discovered in Web data are less structured and consistent than those entered by users on a map application, we expect that the vast amount of data will compensate for the noise.

In the case of Zheng et al., their system is restricted to the city of Beijing, whereas we consider any place in the world. Finally, Zheng et al. populate their list of points of interest with a categorised POI database. Thus users can comment on existing POIs, but no previously unseen POIs are introduced into the system. The focus of their work is to recommend known POIs to users who are in a

given place, which is distinctly different than discovering previously unknown POIs.

Yin et al. [25] do not extract points of interest, but rather model the topics in a given location. In their system a topic is a "spatially coherent meaningful theme". They create a data set based on seven concepts: Landscape, Activity, Manhattan, National Park, Festival, Car and Food, using the topics as keyword to crawl the Flickr API for images associated with those concepts. They propose latent geographical topic analysis to discover sub-topics related to the seven parent concepts. In a second task they identify the regions associated with a given topic. Their system relates to ours in the sense that a POI could be considered a subtopic of a region and a system that is designed to find topic mentions in Flickr data may discover POIs, along with other topics. However, they have constrained their system to regions in the U.S. and allow for the discovery of a wide range of topics.

Our localization component is based on the work of Serdyukov et al., described in [18]. We discuss this work in more detail in Section 6. In similar work, Crandall et al. [6] propose a system to predict among ten landmarks in a given city, within 100 meters, in Flickr images. Their experiments are limited to a specific set of landmarks in a fixed set of cities, as there are no images in their test or training sets that represent places outside of this set of locations. This differs significantly from our task, as we are trying to predict the location of a landmarks and other POIs, anywhere in the world. Furthermore, their work focuses primarily on images, and leverages image features, whereas we work entirely with text.

Yi et al. [24] use language modeling to determine the locations implicit in queries. They use Placemaker to identify location mentions in queries, which they then remove. The resulting queries are intended to contain implicit locations. However, the way in which they use Placemaker is likely to leave mentions of neighborhoods and POIs, as they remove only the primary locations in the queries, in the case that there is more than one location mentioned. This represents an explicit mention of a location, rather than an implicit one. Furthermore, their evaluation is limited to predicting locations that exist in Placemaker. They do not create an independent ground truth.

The work of Hollenstein and Purves [9] seeks to identify vernacular regions in Flickr[7] data. Vernacular regions include mentions such as "Downtown" or "CBD", which are not as granular as POIs, but are significantly smaller than cities, and represent non-official locations that are not typically included in gazetteers. They present a case study of six cities in the U.S. and Europe. This is one of the few studies that attempts to identify regions smaller than a city.

Other work that does not seek to localize geographic entities, but rather to assign a geographic scope to a document includes work to build location topic models from blog data [13, 23], and finding the geographic focus of web pages [7, 4, 27].

# 3. SEQUENTIAL TAGGING MODEL

Named-entity recognition has been well-studied for a number of years. The Message Understanding Conference [1] (MUC) ran from 1987 to 1999. More recently the Automatic Content Extraction Program (ACE) [2] ran from 2000–2008. The CoNLL Shared Tasks for 2002 [19] and 2003 [20] provides a reasonable overview of the standard data and benchmarking tasks for NER.

Conditional Random Fields (CRF) were introduced by Lafferty et al. [11], for text classification and sequence labelling. The CRF was proposed for NER by McCallum and Li [12], and we borrow this approach for POI detection. They report identifying location mentions in the CoNLL English data set with 87% precision and recall. The locations in the CoNLL data refer only to cities, states and countries and the entire data set is composed of news articles. It represents a much simpler problem, in part because of the nature of the location mentions, and the presence of towns, states and countries in gazetteers and because the news data is more semantically rich and cleaner than social media data and Web data.

The conditional random field computes the probability of a label sequence, $y$, given an observation sequence $x$, according to:

$$p(Y|X,\lambda) = \frac{1}{Z(X)} \exp\left(\sum_j \lambda_j F_j(Y,X)\right) \quad (1)$$

where $Z(X)$ is a normalising factor, and $F(Y,X)$ is the set of feature functions computed over the observations and the label transitions. The learning process selects the set of feature weights $\Lambda$ which maximise the label sequence probability $P(Y|X)$:

$$argmax_\Lambda \left\{ \frac{1}{Z(X)} \exp\left(\sum_j \lambda_j F_j(Y,X)\right)\right\} \quad (2)$$

More information about CRFs are given in Wallach [22]. We use Okazaki's implementation of the CRF [16] from the CRFsuite project[8] as it has been shown to outperform other implementations in terms of training speed for sparse data and implements a number of training methods capable of handling large amounts of data - an important consideration for a system designed to learn over large-scale web snippet data. We use the Averaged Perceptron training method [5] with a maximum number of iterations of 10. We limited the iterations in training because it made training over large amounts of data tractable. Our data is labelled in BI0 notation, as is standard for NER tasks; that is, a token is labeled as the beginning of a POI mention (B), the continuation of a POI mention (I), or not part of a POI mention (0).

## 3.1 Features

Each example sentence in our data is treated as a sequence of tokens, represented by a vector of binary features (described in Table 1). The observation features fall into one of four classes: lexical, geographic, grammatical and statistical. Lexical features are computed over the surface text of the token stream.

Geographic features were computed using Yahoo! Placemaker[9], a geographic parsing service, to provide data for tokens that match a POI name. For a token that matches, Placemaker provides information that includes a list of candidate places to which the token may refer and for each,

[7] www.flickr.com

[8] `http://http://www.chokkan.org/software/crfsuite/` visited January 2012

[9] http://developer.yahoo.com/geo/placemaker/ visited January 2012

contextual information like name variants in different languages, and colloquial names. Characterising statistics are computed over this list.

To encode the grammatical function of each token, part-of-speech tagging was done for each token within a sentence using the Apache OpenNLP[10] implementation of a max-ent POS tagger, using the Penn English Treebank POS tag dictionary[11] that comprises of 36 tags.

Normalized pointwise mutual information ($npmi$) was computed over token bigrams appearing in the mobile search query logs of a commercial search engine[12]. For each bigram, the normalised point-wise mutual information of a token $x$ and its subsequent token $y$ was computed as:

$$\begin{aligned} pmi(x;y) &\equiv log\frac{p(x,y)}{p(x)p(y)} \\ npmi(x;y) &= \frac{pmi(x;y)}{-log\left[max\left(p(x),p(y)\right)\right]} \end{aligned} \quad (3)$$

To convert the ($npmi$) into a binary feature, the output values were discretized by applying a greater-than threshold test at each 0.1 interval between -1 and +1, resulting in 20 binary features per bigram.

For the state transition features, we consider the previous state and the next state for all features, except for the word identity and word shape features, which are computed over the previous two, and the next two states (this helps in the common case of longer formulaic POI names such as "Church of Saint Martin" of "the Museum of Natural History").

# 4. POI EXTRACTION

For the extraction task, we work with two data sets: we created a manually annotated data set and a data set of POIs from Wikipedia. The manual annotated data was composed primarily of news articles from the U.S. and the U.K., but also included a small number of examples from Yahoo! Answers[13] and a small number of queries submitted to a search engine. The POIs represented in this data include businesses, services, landmarks and public buildings such as schools, hospitals, airports and prisons.

Our data was annotated by two assessors, both native English speakers (annotating data in English), one from the U.S. and one from the U.K. They were shown random examples from multiple sources, and were instructed to highlight all locations in the text. The inter-assessor agreement was 73.9%. In total 1,337 of the examples they annotated contained POIs, which yielded 1,066 unique POIs.

In addition to measuring the inter-assessor agreement, we measured the precision, recall, and F-measure of one assessor, using the other assessor's data as the ground truth. The results of this is shown in Table 2. This provides a reasonable upper bound on the performance of the POI detection. The results shown in Table 2 are much lower than those reported on the named-entity recognition benchmarking tasks, where the location mentions are entirely composed of cities, states, and countries, and all of the data is news data. The results indicate that identifying POIs in data is not straightforward, even for human assessors. We trained a model using this data with 10-fold cross validation and a 90/10 split. The results are shown in Table 3 in the row labeled "Manual Annotations". The performance of the model is almost as good as that of the human assessors on the same task.

| Ground Truth Assessor | Precision | Recall | F-Measure |
|---|---|---|---|
| 1 | 0.749 | 0.792 | 0.770 |
| 2 | 0.814 | 0.716 | 0.762 |

**Table 2: The result of identifying location boundaries, if one assessor is used as the ground truth labelling and the other assessor is used as the test labelling.**

To create the Wikipedia data set, we selected pages in Wikipedia whose topic was a POI. We determined which Wikipedia pages relate to POIs as follows. The Geonames[14] database encodes geographic entities with a feature code that classifies entities according to an entity taxonomy. These codes are grouped into 9 classes, labelled with a class code letter. The Yago2 ontology gives a concordance between Wikipedia articles and Geonames geographic entities [8]. We selected English language Wikipedia articles which have been identified with the Geonames "S" class, which corresponds most closely to the definition of POIs used in this paper, containing entities such as airports, buildings, facilities and historical and industrial sites. From this set of articles, the title text is used as a surrogate for the name of the POI. The abstract of the article is segmented into sentences and filtered for those that contain the POI name. This process gave us 2,896 unique POIs with a total of 5,186 examples of their use in context.

The results of training a model on the Wikipedia data (with 10-fold cross-validation) are better than the results for the manually annotated data. This reflects the lack of noise in the data, and the use of official names for the POIs. The results are shown in Table 3 in the row labeled "Wikipedia Article Sentences".

The baseline results (in Table 3 , the rows labeled "Placemaker Baseline") were obtained by processing the two data sets with the Placemaker service to extract points of interest mentioned in the text. We expanded the valid place types returned by Placemaker to include airports, and land features which are not POIs in the Placemaker classification, but are in the Geonames classification.

It is important to note that the Wikipedia data is labeled according to whether it is a POI or some other type of entity, but it has not been labeled with sequence labeling (such as the BI0 notation described above) that would be needed for NER. For the purpose of detecting mentions of POIs with a sequential tagger, the data is unlabeled. Thus, this result represents a first system to recognize geographic points of interest, entirely from unlabeled data. Both the system trained on manual annotations, and the system trained on Wikipedia data performed significantly better than the Placemaker baseline.

# 5. BOOTSTRAPPING FROM THE WEB

[10] http://incubator.apache.org/opennlp/ visited February 2012

[11] `http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html` visited January 2012

[12] details removed to preserve the anonymity of the authors.

[13] `http://answers.yahoo.com/` visited February 2012

[14] `http://www.geonames.org/` visited January 2012

| Feature | Description |
|---|---|
| Word Identity | The raw text representation of the token |
| Normalised Word Identity | The lower case version of *Word Identity* |
| Word Shape | Indicates capitalisation, and hyphens |
| Word Capitalisation | The first letter of the token is a capital letter |
| Word Position (First) | The token is at the beginning of a sentence |
| Word Position (Last) | The token is at the end of a sentence |
| Word Prefix | First three characters of the token |
| Word Suffix | Last three characters of the token |
| Part-Of-Speech | OpenNLP English language max-ent labelling |
| Bi-Gram | Normalised point-wise mutual information of token and next token |
| Related Location Probability | Probability that token represents a place |
| Related Location Match | True if token matches a place name |
| Related Location Size | Number of place matches including variants |
| Related Location Unique | Place matches where variants are conflated |
| Related Location Unique Ratio | (*Related Location Size*)/(*Related Location Unique*) |

**Table 1: Lexical, grammatical, statistical and geographic features used by the CRF tagger.**

| Data Set | Precision | Recall |
|---|---|---|
| Placemaker Baseline Manual Annotations | 0.238 | 0.233 |
| Manual Annotations (10-fold c.v.) | 0.686 | 0.467 |
| Placemaker Baseline Wikipedia | 0.133 | 0.209 |
| Wikipedia Article Sentences (10-fold c.v.) | 0.872 | 0.742 |

**Table 3: Results of training and evaluating on the data using 10-fold cross-validation. The baseline is the result of using Placemaker to identify POIs in the manual annotations data, and the Wikipedia data. The systems trained on either manual annotations, or Wikipedia, give significantly better results than Placemaker, which is the state-of-the- art.**

Both the Wikipedia and the manual annotation data sets are very small, which means geographic coverage, as well as their coverage in terms of the types of POIs mentioned is limited. In addition, the mentions of POIs vary greatly from one data source to another. For example, the University of Buffalo might be referred to as **#UBuffalo**, or **University of Buffalo** depending on the data source and the context of the mention. Creating enough manually annotated data to learn patterns from this amount of variation would be a major undertaking. In this section we show that increasing the amount of training data by bootstrapping from the Web yields a significant improvement in the learned POI extraction.

## 5.1 Bootstrapping Data

The Wikipedia title text was used as seed queries to the Bing search engine via their web-based API[15] to retrieve snippets or web page abstracts relevant to those queries. We retrieved up to 10 web snippets per Wikipedia title. The snippets provide a small amount of text to contextualize the POI. The idea is to provide a context in which a POI is used, to enable the model to learn a more general representation of the POI. The resultant list of POIs from Wikipedia is relatively clean, but there is no guarantee that the POI will be mentioned in the proper context in the search engine snippets. For example in this scenario the POI "The White House" might retrieve a web snippet about white houses. This process gave data for 2,896 total unique POIs and 21,228 examples of their use in context.

As stated earlier, the POIs mentioned in Wikipedia are largely mentions of permanent structures such as landmarks and government buildings, usually represented by their official name. For Web applications, the definition of POI also includes more ephemeral places such as restaurants and local businesses. Location check-in services such as Foursquare[16] and Gowalla[17] generate a large number of such POIs. An advantage of this data is that it has high coverage of places the users of these applications actually visit.

Both Foursquare and Gowalla provide public APIs that allow their data to be crawled, within a rate limit. The POIs in this data consist largely of mentions of businesses, but also include landmarks and public buildings such as libraries. Users may select from lists of known POIs (mostly sponsored listings, or licensed data), or they may create their own POI. The majority of check-ins in our data are to pre-existing POIs. They are relatively clean, because the formulaic way in which they appear in the data allows them to be extracted reliably. Although the POIs in this data could be used to create a lexicon of POIs, the POI check-ins cannot be used directly to train or evaluate a sequential model because they contain no textual context.

Once the elements that represent POI names are extracted from the checkin, they are used as seed queries to the Bing search API. These snippets contain sentences where the POI has been used in context (as opposed to the terse, formulaic mentions in the checkins). It should be noted however that

[15] `http://www.bing.com/toolbox/bingdeveloper/` visited February 2012

[16] `http://www.foursquare.com/` visited October 2011

[17] `http://www.gowalla.com/` visited October 2011

| Dataset | Total Sample Sentences | Total POIs | Avg. Sentences per POI |
|---|---|---|---|
| Manual Annotations | 1,337 | 1,066 | 1.25 |
| Wikipedia Article Sentences | 5,186 | 2,896 | 1.79 |
| Wikipedia Bootstrapped Snippets | 21,228 | 2,896 | 7.33 |
| Gowalla Web Snippets | 50,000 | 40,152 | 1.25 |
| Foursquare Web Snippets | 50,000 | 47,858 | 1.04 |

**Table 4: Characteristics of the experimental data sets**

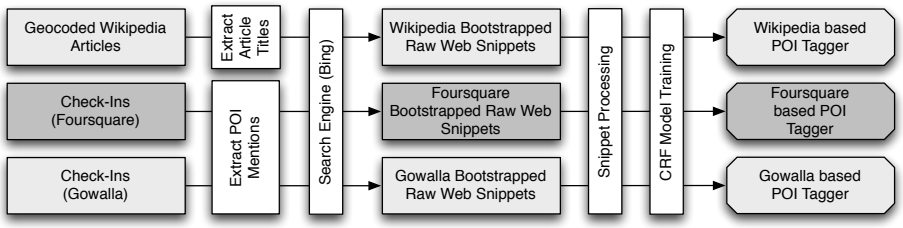


**Figure 1: The process to generate a sequential tagger involves extracting mentions of POIs in social media and using these as seed data to generate web snippets.**

| Source Data | Precision | Recall |
|---|---|---|
| Foursquare | 0.058 | 0.044 |
| Gowalla | 0.075 | 0.054 |
| Wikipedia | 0.153 | 0.182 |

**Table 5: Results of evaluating each web snippet based data set using Yahoo! Placemaker**

this mechanism does not guarantee the snippets' relevance to the seed POI query. Nor does it ensure that all possible POIs present in web snippets are correctly labelled.

This process was carried out for all Foursquare and Gowalla check-ins, with 10 snippets being retrieved from each seed POI query. This generated millions of sample sentences that could be used for training. For each check-in service, we randomly sampled a subset of 50,000 examples, giving the data set characteristics shown in Table 4. These samples were selected with the criteria that they contained the POI as an exact substring and that the sample contained only unextended ASCII characters. Table 5 shows the bootstrapped snippets tagged for POIs using Placemaker, which represents the state-of-the-art in POI detection. The baseline for the manual annotations was shown in Table 3.

The results of training and testing on this data using 10-fold cross validation, as well as training on the checkin data and evaluating on the manual annotations are shown in Table 6. Note that bootstrapping the data improved the results for the checkin data, whereas the bootstrapped Wikipedia results are much lower than simply training on the original Wikipedia POI mentions.

## 6. POI LOCALIZATION

In order for the discovery of a point of interest to be useful in an application, it must be associated with a location. We employ the location modeling approach proposed by Serdyukov et al. [18] to predict the location of the POIs in a sample of the manually annotated data set. We evaluate our approach in terms of the distance in kilometers from the ground truth location of each POI. The Placemaker service is our baseline, as it is the current state-of-the-art. Finally, we use our model and Placemaker in a cascade architecture to improve the results over either system alone.

The location models described in [18] are built by quantizing the coordinate system into one kilometer, 10 km or 100 km cells. For the work in this paper, we choose one kilometer grid cells because we are predicting geographic entities that are mostly smaller than one square kilometer. As in [18], each cell is associated with the geo-tagged Flickr images that were taken within the cell boundaries. We estimated the models from the raw tags associated with ten million geotagged Flickr images, uploaded to Flickr before October 2010.[18] The cell is represented by the distribution of tags associated with its images. The problem of predicting a location can then be reduced to a standard information retrieval ranking problem where cells are "documents" and the image tags are the terms in the document. In [18] the terms in the cell-documents are weighted by their term frequency. This produces a model that might be biased toward the tags applied to a set of images by a single user. To remediate this, in our models we weight the terms in the cell-documents according to the user frequency, the number of users who have applied that tag to that cell, rather than the number of times the tag has been applied in that cell. More specifically,

$$P(t|\theta_L) = \frac{c_{user}(t, L)}{|L|}, \tag{4}$$

where $c_{user}(t, L)$ is the number of unique users who use the term in the location (cell). $|L|$ is calculated as the sum of the user frequency of all terms in the location:

$$|L| = \sum_{t_i \in L} c_{user}(t_i, L). \tag{5}$$

Weighting the terms according to the user frequency, rather

[18]http://www.flickr.com visited February 2012. Flickr also has a public API that allows the data to be crawled.

| Training Data | Testing Data | Precision | Recall |
|---|---|---|---|
| Yahoo! Placemaker | All Manual Annotations | 0.2372 | 0.2281 |
| Wikipedia † | All Manual Annotations | 0.514 | 0.337 |
| Wikipedia | Known Manual Annotations | 0.447 | 0.397 |
| Wikipedia | New Manual Annotations | 0.521 | 0.324 |
| Foursquare † | All Manual Annotations | 0.276 | 0.655 |
| Foursquare | Known Manual Annotations | 0.215 | 0.735 |
| Foursquare | New Manual Annotations | 0.288 | 0.638 |
| Gowalla † | All Manual Annotations | 0.360 | 0.414 |
| Gowalla | Known Manual Annotations | 0.314 | 0.510 |
| Gowalla | New Manual Annotations | 0.362 | 0.393 |
| Wikipedia | (10-fold c.v.) | 0.879 | 0.955 |
| Foursquare | (10-fold c.v.) | 0.689 | 0.468 |
| Gowalla | (10-fold c.v.) | 0.857 | 0.868 |

**Table 6: Results of POI tagger trained on bootstrapped web snippet data, evaluated on both human annotated data and using cross validation (c.v.). When each of the three trained models (marked with †) are compared with the baseline Yahoo! Placemaker evaluation, they are found to be significantly different with p-value $<$ 0.001 according to McNemar's $\chi^2$ test.**

than the term frequency, reduces the effects of bulk uploading, or of applying near-duplicate tag sets to multiple images. It has been proposed before for choosing representative tags for a given location [3, 10], and for suggesting tags for photos [14].

To evaluate the system, we created a ground truth data set. Although the coordinates given by Placemaker for POIs are derived from curated data sources, and are as accurate as reasonably can be expected, Placemaker does not identify every POI in the data, and when comparing Placemaker to other approaches, both must be compared to a common ground truth. We sampled 400 examples from the manually annotated data described in Section 4. The ground truth locations of the POIs were determined by identifying the address (or geographic coordinates when they were available) of the POI from its official web page. We located the address on a map, and the POI was verified visually by zooming in on the satellite view of the map. POIs were discarded in cases where the POI could not be visually verified on the map, or where the exact address of the POI was not available on its official web page. In our data, out of 400 examples of POIs, 291 represented unique locations according to the geographic coordinates, and 327 were unique mentions of POIs.

We measured the Vincenty Distance [21] in kilometers from the predicted location of the POI to its true location. Our localization method predicts the centroid of a one-kilometer grid cell, so our system can be at most accurate within 500 meters. We use the median distance as the metric instead of the average, because a single location that is predicted incorrectly to be on the opposite side of the globe will skew the average distance in such a way as to make the results difficult to interpret.

Our evaluation considers three subsets of the data. The first consists of the subset of examples which are identified by Placemaker as points of interest. We call these "known" locations. The second is the set of examples that Placemaker identifies as some other type of location (such as a city, or a country - the "other" locations). The third consists of locations identified by Placemaker as not containing a location. We call these "new" locations, although they are only new in the sense that they are not in the curated data used by Placemaker. Table 7 shows that placemaker is able to identify 265 locations out of 400, of which 134 are identified as POIs, and 131 are identified as other types of locations. This represents a recall of roughly 33%. Our system localizes 395 out of 400 POIs, a recall of roughly 98%.

| | 265 Locs | 134 POIs | 131 Other |
|---|---|---|---|
| Placemaker | 1.17 | 0.29 | 4.19 |
| Location Model | 1.77 | 0.72 | 3.45 |
| Cascade Model | 0.82 | 0.29 | 2.90 |

**Table 7: The distance, in kilometers, from the true location to the location predicted by the experimental systems. The 265 locations included in this table were identified by Placemaker, out of 400 total examples. Of the locations found by Placemaker, 134 were identified by Placemaker as points of interest, as opposed to other types of locations such as cities, or states.**

When Placemaker identifies a point of interest in text, it localizes it with very high accuracy (roughly 300 meters), thus it makes sense to use Placemaker to find as many POIs as it can. In the cascade model, Placemaker is used first to identify any POIs in the data. For examples that Placemaker does not find a POI, if it finds a location of another type, the bounding box of that location is used to constrain the search for the location in the Location Model, by taking the first result in the ranked list of results returned by the Location model that falls within the bounding box of the location returned by Placemaker.

The Location Model performance in localizing POIs that Placemaker identified as other types of locations is an improvement over the Placemaker result. Using the location information returned by Placemaker, even when Placemaker fails to identify the POI, improves the result further.

For the 130 POIs localized by our system that were determined to contain no place mention by Placemaker, the median distance from the true location was a somewhat depressing 439 kilometers. Clearly, having location information to constrain the search improves performance. The Placemaker system was designed to provide information about

| | Placemaker | Cascade Model | Geo Scope Model | Number of Examples |
|---|---|---|---|---|
| Placemaker POIs | 0.29 | 0.29 | 0.29 | 134 |
| Placemaker Other Locations | 4.19 | 2.90 | 2.12 | 131 |
| All Known Locations | 1.17 | 0.82 | 0.79 | 265 |
| New Locations | – | 439.0 | 5.88 | 130 |
| All Data | – | 1.20 | 0.96 | 395 |

**Table 8: The distance, in kilometers, from the true location to the location predicted by the Cascade Model, constrained by the geographic scope of the document in which the location mentions appear. The Placemaker results are repeated for the sake of comparison**

the mentions of each individual location in free text, but also to provide information about the text as a whole, such as its geographic scope (the minimum bounding box enclosing most of the locations mentioned in the text). In our data, for the 130 POIs that were deemed non-locations by Placemaker, we took the geographic scope of the news article the POI was originally extracted from, and used it to constrain the search in our system. Table 8 shows the results over all examples of the median distance from the predicted location to the true location, when the search in the Cascade model is constrained by the geographic scope.

For locations that are identified by Placemaker, it is clear that other disambiguators in the POI mention improves performance. For example, in the POI "MGM Grand Garden Arena, Las Vegas, NV" Placemaker is unable to identify the arena as the POI, and instead returns the city of Las Vegas. The Location Model ranks the locations associated with the terms mentioned in the POI. In the Cascade Model, after ranking the locations by their score, the list is scanned for the locations that are contained in the bounding box for Las Vegas, NV, and the first location within the bounding box is returned as the correct location of the POI. In the Geo Scope Model, rather than return a bounding box for "Las Vegas, NV", Placemaker returns the geographic scope of the article the POI mention originally appeared in to constrain the search. It is important to note that there is no guarantee that a given POI in an article will be within the geographic scope of the article.

# 7. DISCUSSION

This work has identified the significant difference between traditional location NER and the task of POI recognition. Such recognition is non-trivial, as demonstrated in Section 4, in which we showed that even with POI labelling undertaken in a strictly controlled environment, consensus was difficult among human assessors.

The subjectivity of POIs also highlights their diverse nature and how their definition is closely dependent on the application in which they are used. It also makes it hard to generate sufficient quantities of manually annotated data for training robust models.

To tackle these problems, we first used Wikipedia as a source of POI mentions and their usage in context. This produced models that performed well at recognizing POIs in Wikipedia articles. However, the POIs found in such text do not reflect the range and variety of those that are used in location-based social media and mobile applications, such as businesses and restaurants.

Social media data, such as location checkins, are an extensive and easily accessible source of POIs, but they lack the textual context required for training sequential models. Extending POIs mentioned in social media with web search engine snippets improves performance over the state of the art. The data itself is not manually labeled, so there is the potential to train on vast amounts of this data cheaply. Furthermore, by being an entirely automatic process run on dynamic data from the Web, models can be continually trained and updated, to capture ephemeral and temporary POIs.

We see that amplifying the Wikipedia data with web snippet data degrades performance. Wikipedia is a very clean data source, and the POIs mentioned in Wikipedia are usually in their canonical form, and in the proper context. Although the data set is small, it is sufficient to predict the entity boundaries for these POIs. It is important to note that such a system can be trained from unlabeled data. Adding web snippet data, however, only makes the data noisier, without adding information.

By contrast, the checkin data benefits greatly from bootstrapping with web snippets. This type of data is an excellent source of POI mentions, but even when there is textual context surrounding the POI mention in the checkin, it is not sufficiently informative to estimate a model from it. Other types of social media data, such as Twitter, may be a good source of information about places that people visit, but it is extremely noisy, and contains abbreviations and textual shortcuts inherent to the constraints of the application itself. Bootstrapping from the web allows the POIs to be placed in a natural language context, which, while noisy, is considerably less noisy than the Twitter data itself.

One reason for lower results on large amounts of data compared to the smaller data sets from Section 4 is the variation in the POIs themselves. They represent different classes of entities and it may be necessary to learn them as distinct classes. It is possible that a mention of a local business in text is not sufficiently similar to a mention of a landmark or library or public school. Since we did not canonicalize the POI mentions or aggregate them according to their geographic coordinates, we do not have equivalences among the POIs. Since the evaluation relies on an exact match of the POI entities, the system is penalized for correct mentions that do not match exactly.

With regard to localization, we have shown that we can determine the location of the POI with high accuracy, less than one kilometer for most POIs. In the evaluation, we did not evaluate the end-to-end performance of the entire system, because if a set of tokens is erroneously tagged as a POI, it is not meaningful to try to localize it. The system may correctly guess the location, but if the input to the system is not valid, the evaluation will not be meaningful. The evaluation on the ground-truth tagging gives an upper-bound on the localization performance, as errors generated

in the tagging phase will propagate to the localization phase. It may be possible to leverage the localization, either as a filter step, or as a feature in the tagger, to improve the performance of the tagging, but this is left to future work.

The localizer is built on Flickr data, specifically tag sets. The tag sets in Flickr are unique in that they contain mostly nouns, and a few adjectives, and very little noise. Many of the images that are geotagged and uploaded to Flickr portray POIs and tourist destinations. This makes Flickr ideal for localizing POIs in the places where Flickr has coverage. For locations with no Flickr coverage, it is not possible to predict the location. This is a limiting factor that affects all systems built on social media.

The Placemaker system localizes POIs within a median distance of 1.1 kilometers for the locations it identifies (which is just over half of the examples). For the POIs that Placemaker identifies as such, the median distance from the ground truth is around 300 meters. This gives a reasonable bound on the performance of any localization system because the Placemaker system is built upon surveyed, curated data. The differences in the distance from the ground truth can partially be accounted for by the fact that some POIs (such as airports, and university campuses) are larger than one kilometer, and it is not clear how to express the location with a latitude/longitude point.

# 8. CONCLUSIONS

Points of interest form the basis of content for a growing number of mobile and social media applications. Local search and recommender systems rely on knowing the points of interest in a city in order to understand a user's geographic context, to better serve relevant results. Automatically detecting POIs allows us to develop systems that are dynamic and that reflect the services and places people visit in a city in the course of their daily lives. In this paper we presented a system for detecting POIs in unstructured text from labelled and unlabelled data. We showed system performance on manually annotated data composed mostly of news articles, on Wikipedia articles and on mentions of POIs bootstrapped from social media. We can achieve a precision of up to 87% training and testing on unlabelled data. For half of the POIs, we can identify their location within one kilometer of the true location.

For future work we would like to introduce a more lenient evaluation metric that allows for small variations in the name of a POI. We also intend to refine the learning algorithm to leverage social media data more effectively and to incorporate physical information about the POI such as its relationship to the geography in which it is situated. Finally, as the research community builds more systems on social media data, the credibility of this data must be better understood to determine which data examples are most suitable for training and evaluation.

# 9. REFERENCES


[1] `http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html` visited October 2011.

[2] `http://www.itl.nist.gov/iad/mig//tests/ace/` visited October 2011.

[3] S. Ahern, M. Naaman, R. Nair, and J. Yang. World Explorer: Visualizing aggregate data from unstructured text in geo-referenced collections. In *JCDL '07*, 2007.

[4] E. Amitay, N. Har'El, R. Sivan, and A. Soffer. Web-a-where: geotagging web content. In *SIGIR '04*, pages 273–280, New York, NY, USA, 2004. ACM.

[5] M. Collins. Ranking algorithms for named-entity extraction: Boosting and the voted perceptron. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*, 2002.

[6] D. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *Proceedings of the 18th International Conference on World Wide Web*, pages 761–770. ACM, 2009.

[7] J. Ding, L. Gravano, and N. Shivakumar. Computing geographical scopes of web resources. In *VLDB '00*, pages 545–556, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[8] J. Hoffart, F. Suchanek, K. Berberich, E. Kelham, G. de Melo, G. Weikum, F. Suchanek, G. Kasneci, M. Ramanath, and A. Pease. Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Commun. ACM*, 52(4):56–64, 2009.

[9] L. Hollenstein and R. Purves. Exploring place through user-generated content: using Flickr to describe city cores. *Journal of Spatial Information Science*, (1), 2010.

[10] L. Kennedy, M. Naaman, S. Ahern, R. Nair, and T. Rattenbury. How flickr helps us make sense of the world: context and content in community-contributed media collections. In *Proceedings of the 15th international conference on Multimedia*, MULTIMEDIA '07, pages 631–640, New York, NY, USA, 2007. ACM.

[11] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pages 282–289, 2001.

[12] A. McCallum and W. Li. Early results for named entity recognition with conditional random fields, feature induction, and web-enhanced lexicons. In *Proceedings of CoNLL*, 2003.

[13] Q. Mei, C. Liu, H. Su, and C. Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *WWW '06*, 2006.

[14] E. Moxley, J. Kleban, and B. S. Manjunath. Spirittagger: a geo-aware tag suggestion tool mined from flickr. In *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, MIR '08, pages 24–30, New York, NY, USA, 2008. ACM.

[15] L. Mummidi and J. Krumm. Discovering points of interest from users' map annotations. *GeoJournal*, 72:215–227, 2008.

[16] N. Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs), 2007.

[17] D. Segal. "Closed, Says Google, but Shops' Signs Say Open". *The New York Times*, September 5, 2011.

[18] P. Serdyukov, V. Murdock, and R. van Zwol. Placing Flickr Photos on a Map. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 484–491. ACM, 2009.

[19] E. F. Tjong and K. Sang. Introduction to the conll-2002 shared task: Language-independent named

entity recognition. In *COLING-02 proceedings of the 6th Conference on Natural Language Learning*, 2002.

[20] E. F. Tjong, K. Sang, and F. de Meulder. Introduction to the conll- 2003 shared task. In *CoNLL '03 Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 2003.

[21] T. Vincenty. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey Review*, XXIII(176), April 1975.

[22] H. Wallach. Conditional random fields: An introduction. Technical Report Technical report MS-CIS-04-21, University of Pennsylvania, 2004.

[23] C. Wang, J. Wang, X. Xie, and W.-Y. Ma. Mining geographic knowledge using location aware topic model. In *GIR '07*, 2007.

[24] X. Yi, H. Raghavan, and C. Leggetter. Discovering users' specific geo intention in web search. In *WWW '09: Proceedings of the 18th International Conference on World Wide Web*, pages 481–490, New York, NY, USA, 2009. ACM.

[25] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang. Geographical topic discovery and comparison. In *Proceedings of the 20th International World Wide Web conference (WWW'11)*, 2011.

[26] V. W. Zheng, Y. Zheng, X. Xie, and Q. Yan. Collaborative location and activity recommendation with gps history data. In *Proceedings of the 19th International World Wide Web conference (WWW'10)*, 2010.

[27] W. Zong, D. Wu, A. Sun, E.-P. Lim, and D. H.-L. Goh. On assigning place names to geography related web pages. In *JCDL '05*, pages 354–362, New York, NY, USA, 2005. ACM.