A Weighted Multi-Attribute Method for Matching User-Generated Points of Interest

Grant McKenzie*[1], Krzysztof Janowicz†[1] and Benjamin Adams‡[2]

[1]Department of Geography, The University of California, Santa Barbara, USA
[2]Centre for eResearch, Department of Computer Science, The University of Auckland, New Zealand

Abstract

To a large degree, the attraction of Big Data lies in the variety of its heterogeneous multi-thematic and multi-dimensional data sources and not merely its volume. To fully exploit this variety, however, requires conflation. This is a two step process. First, one has to establish identity relations between information entities across the different data sources; and second, attribute values have to be merged according to certain procedures which avoid logical contradictions. The first step, also called matching, can be thought of as a weighted combination of common attributes according to some similarity measures. In this work, we propose such a matching based on multiple attributes of Points of Interests (POI) from the Location-based Social Network Foursquare and the Yelp local directory service. While both contain overlapping attributes that can be use for matching, they have specific strengths and weaknesses which makes their conflation desirable. For instance, Foursquare offers information about user check-ins to places, while Yelp specializes in user-contributed reviews. We present a weighted multi-attribute matching strategy, evaluate its performance, and discuss application areas that benefit from a successful matching. Finally, we also outline how the established POI matches can be stored as Linked Data on the Semantic Web. Our strategy can automatically match 97% of randomly selected Yelp POI to their corresponding Foursquare entities.

Keywords: Points of Interest; Matching; Linked Data; Volunteered Geographic Information

Introduction

Recently, an economy of data and service providers has evolved around Points Of Interest (POI).[1] This includes map-centric applications such as Google Maps, local directory services such as Yelp, several location-based social networks, e.g., Foursquare, as well as numerous spatially-enabled sharing services such as Path or Flickr. Each of these services specializes on certain kinds of place-related information. Additionally, while they collect user-generated content, access to this content is restricted by APIs, frequency limitations, e.g., 1000 queries per day, as well as storage restrictions, e.g., the queried data has to be deleted within 24 hours. As user data and profiles are the key assets of these companies, they have to carefully balance openness and interlinkage to other platforms with their own interest in walling-off the data.

From a research perspective, however, the combination of these data sources would be desirable for multiple reasons. First, by conflating Points of Interest, we can exploit complementary attributes to arrive at a more holistic understanding of places. For instance, one can combine user reviews from different communities to study sentiment, compare the place categorization hierarchies and match them using ontology alignment techniques, mine check-in behavior for patterns, compare pictures from tourists versus locals, and so on. In fact, this *variety* aspect is one of the key value propositions of Big Data and semantic interoperability in general. Second, we can increase data quality by comparing the same attributes across data sets. One potential application would be to remove typos in place names contributed by volunteers. Finally, combining the data sources would also increase their coverage.

The process of conflating Points Of Interest can be divided into two steps. First, identity has to be established between them. That is, it has to be determined whether both information entities correspond to the same place in the physical world. We refer to this part as *matching* throughout the paper. To do so, one would usually compare the values of attributes common to both datasets using a particular similarity measure. For example, if two datasets both contain a name attribute for their Points of Interest, the Levenshtein distance can be used to match them. Simply comparing names alone, however, will only work for certain cases. Thus, other attributes such as geographic locations and tags will be compared using appropriate measures as well. In practice, these measures will rarely return exact matches, and we have to combine them and define a matching threshold. This, of course, assumes that we can successfully match the attributes in the first place; i.e., establish that *place_name* in one datasets describes the same attribute as *POIname* in another one. Strictly speaking, it also requires an understanding of what it means for places to be equal.

*grant.mckenzie@geog.ucsb.edu

†jano@geog.ucsb.edu

‡b.adams@auckland.ac.nz

[1] We use the term Points Of Interest here instead of the more general Places Of Interest as the used data sources only contain point-like feature geometries.

If an established restaurant moves to another building, does it become a new place even though we still unambiguously refer to it using the same name? Similarly, is an old movie theater that becomes a night club known by the same name, and preserving the theater's original ambiance, still the same place?

In the second step, the attributes of the involved Points of Interest have to be conflated. For example, while a place may have multiple names and one can be chosen to be canonical, this is not feasible for geographic locations. Understanding how to proceed with different attributes is an ontological question. For instance, a particular POI definition may only allow for one place category such as *Restaurant* (see Equation 1). Consequently, in this example, conflating data may require one to use the least upper bound of a hierarchy of place types or any other method that meaningfully reduces the set of types to one.

$$POI \sqsubseteq Place \sqcap \exists hasName.Name \sqcap (= 1\, hasCategory.Type) \sqcap \ldots \qquad (1)$$

In addition to this ontological perspective, it is also beneficial to understand the process by which attribute values are recorded as well as the resulting types of errors. For example, one could naively assume that because POI locations from location-based social networks (LBSN) are recorded by GPS positioning via smartphones, they may be inaccurate to about 5-30 meters and averaging positions from two LBSN would improve accuracy. We will later discuss why this is not the case.

In this work we will focus on the first step of conflation and show how to match POI from the LBSN Foursquare and the local directory services Yelp. Foursquare specializes on user check-ins and, thus, social and temporal aspects. While it also provides user tips, those are typically short personal statements. In contrast, Yelp focuses on detailed user reviews and a wide range of semi-structured place attributes such as the ambiance, prices, noise level, and wifi availability. This makes conflating POI based on both datasets attractive; see Figure 1 for a comparison. For example, this would enable queries for *places visited by friends, that have a low noise level, friendly staff, and free wifi.* Our more immediate interest, however, lies in exploiting the conflated POI to improve user similarity measures for the analysis of sparse semantic trajectories [McKenzie et al, 2013a].

The contributions of this work are as follows:

- Intuitively, one may assume that both datasets have well curated and canonical place names for their POI. Consequently, a syntactic string measure such as Levenshtein distance should be a strong matcher. We will test whether it can successfully match at least 80% of our sample data. It is important to keep in mind that for an automatic matcher a success rate of 80-90% is not sufficient. In our case, given the > 30 million POI in the USA alone, at least 3 million POI would still have to be corrected and matched manually.

- Following the Pareto principle [Reed, 2001], we assume that matching the remaining (less than) 20% of POI will require a weighted combination of matchers which exploit additional POI attributes. First, we will investigate whether an alternative place name matcher can improve our previous results. To do so, we will use *Double Metaphone* to match for phonetic similarity. Next, we will introduce matchers based on place categories, textual user reviews, as well as geographic distance, and evaluate their performance. To the best of our knowledge, phonetic and user review based matchers have not been used in the literature before.

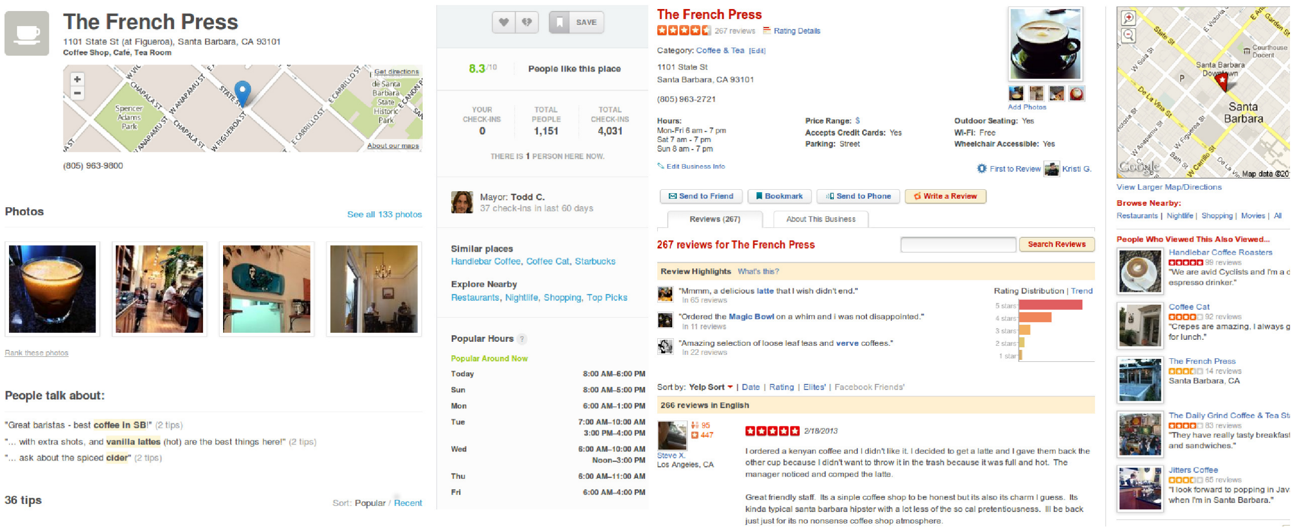
- Subsequently, we will use binomial probit regression to arrive at a weighted combination of all matchers and evaluate the results against an ordinal weight combination and an unweighted base line.

- For the development and evaluation of our matching strategy we selected a subset of all POI so that for each randomly selected Foursquare POI there exists a *true positive* matching POI in Yelp. However, this is not always the case especially if we also take other platforms such as OpenStreetMap, Google Places, Yahoo Local, etc into account. Thus, we will also investigate what match score should be used for the automatic on-the-fly matching of POI from different sources. While essential for matching noisy data like VGI on the Web, the challenging topic of arriving at robust match scores has not been discussed in the literature before.

- Given the restrictions of the used APIs, we outline how to use Linked Data to preserve the match results. This allows to conflate the data on-the-fly in the future without violating the terms of usage.

- Finally, we will discuss some interesting insights made during our work. For instance, we will try to explain why the geographic coordinates of POI clearly differ between Yelp and Foursquare.

The remainder of this paper is divided into 4 sections. First, we will give a brief overview of related work. Next, we will introduce the POI attributes, the matching methodology, and the datasets. We then evaluate our work and compare the performance of the independent measures as well as their combination. Finally, we present conclusions, observations, and directions for further research.



The French Press
1101 State St (at Figueroa), Santa Barbara, CA 93101
Coffee Shop, Café, Tea Room

(805) 963-9800

Photos

See all 133 photos

Rank these photos

People talk about:

"Great baristas - best coffee in SB!" (2 tips)
"... with extra shots, and vanilla lattes (hot) are the best things here!" (2 tips)
"... ask about the spiced cider" (2 tips)

36 tips

Sort: Popular / Recent

8.3/10 People like this place

YOUR CHECK-INS	TOTAL PEOPLE	TOTAL CHECK-INS
0	1,151	4,031

THERE IS 1 PERSON HERE NOW.

Mayor: Todd C.
37 check-ins in last 60 days

Similar places
Handlebar Coffee, Coffee Cat, Starbucks

Explore Nearby
Restaurants, Nightlife, Shopping, Top Picks

Popular Hours ?

Popular Around Now

Today	8:00 AM–6:00 PM
Sun	8:00 AM–5:00 PM
Mon	6:00 AM–1:00 PM
Tue	7:00 AM–10:00 AM 3:00 PM–4:00 PM
Wed	6:00 AM–10:00 AM Noon–3:00 PM
Thu	6:00 AM–11:00 AM
Fri	6:00 AM–4:00 PM

The French Press
267 reviews Rating Details

Category: Coffee & Tea [Edit]

1101 State St
Santa Barbara, CA 93101

(805) 963-2721

Hours:
Mon-Fri 6 am - 7 pm
Sat 7 am - 7 pm
Sun 8 am - 7 pm

Price Range: $
Accepts Credit Cards: Yes
Parking: Street

Outdoor Seating: Yes
Wi-Fi: Free
Wheelchair Accessible: Yes

First to Review Kristi G.

267 reviews for The French Press

Review Highlights What's this?

"Mmmm, a delicious latte that I wish didn't end."
In 65 reviews

"Ordered the Magic Bowl on a whim and i was not disappointed."
In 11 reviews

"Amazing selection of loose leaf teas and verve coffees."
In 22 reviews

Sort by: Yelp Sort | Date | Rating | Elites' | Facebook Friends'

266 reviews in English

Steve X.
Los Angeles, CA
95
447

2/18/2013

I ordered a kenyan coffee and I didn't like it. I decided to get a latte and I gave them back the other cup because I didn't want to throw it in the trash because it was full and hot. The manager noticed and comped the latte.

Great friendly staff. Its a sinple coffee shop to be honest but its also its charm I guess. Its kinda typical santa barbara hipster with a lot less of the so cal pretentiousness. Ill be back just just for its no nonsense coffee shop atmosphere.

View Larger Map/Directions

Browse Nearby:
Restaurants | Nightlife | Shopping | Movies | All

People Who Viewed This Also Viewed...

Handlebar Coffee Roasters
99 reviews
"We are avid Cyclists and I'm a d
espresso drinker."

Coffee Cat
92 reviews
"Crepes are amazing, I always g
for lunch."

The French Press
14 reviews
Santa Barbara, CA

The Daily Grind Coffee & Tea St
83 reviews
"They have really tasty breakfast
and sandwiches."

Jitters Coffee
65 reviews
"I look forward to popping in Jav
when I'm in Santa Barbara."

Figure 1: The Santa Barbara *French Press* on Foursquare (left) and Yelp (right).

Related Work

The matching and conflation of geographic datasets has a long history in the field of geographic information science. Work in this area has typically divided conflation and matching patterns by the attributes present and the end goal of the research being conducted. Historically, two related areas of research have emerged, one focusing on the geometric or geographic properties of the data [Chen et al, 2006; Devogele, 2002; Haunert, 2005; Li and Goodchild, 2011] and another centered on the descriptive attributes [Hastings, 2008]. Adams et al [2010] proposed a general framework for conflation that combines geometric and other attributes. Likewise, work by Sehgal et al [2006] integrated spatial and non-spatial components (names, types and demographic information) of geospatial locations with the goal of consolidating a collection of *true* locations. Recent work on online social networking applications has proposed matching Qype and Facebook Places to OpenStreetMap POI through geographic distance and name matching [Scheffler et al, 2012]. While related in nature, the number of attributes accessible was limited, restricting the ability of the researchers to explore a weighted approach.

Matching attributes of spatial data can take any number of forms, though it usually involves some level of place name matching. The concept of name matching has been investigated in several different computationally focused fields. Hundreds of methods have been developed for analyzing text and assessing similarities between strings for duplication detection [Bilenko and Mooney, 2003; Elmagarmid et al, 2007; Lait and Randell, 1996], language translation [Freeman et al, 2006], and information retrieval [Cohen et al, 2003; Jones and Purves, 2008], many of which have resulted in patents [Lamburt et al, 2002; Page, 2001]. Though name matching is a common technique used in matching and conflating POI, the geographic coordinates of the POI also play a significant role. Research by Wu and Winter [2009] focused on the semantic issues involved in matching place names in a gazetteer. They found that the spatial properties of an entity could be engaged as a supplemental source for matching. Similarly, Mülligann et al [2011], utilized the spatial-semantic interaction of point features to determine duplicates in OpenStreetMap.

Hastings [2008] explicitly took a multi-attribute approach to conflating digital gazetteers focusing on geospatial, geotaxial, and geonominal metrics associated features. His work explored the taxonomy of place types such as *lakes* or *marshes* commonly associated with a specific geographic category. Instead, this work builds on our previous work [McKenzie et al, 2013b] focusing on categorical descriptions of locations based on the activities afforded by the POI as well as the descriptive text contributed by individuals.

The comparison of documents based on unstructured text has been an area of significant research in the past few years. Recent advancements in semantic analysis [Joachims, 1998] and probabilistic topic models [Blei et al, 2003; Ramage et al, 2009] have made it feasible to infer and measure similarities between documents. These topic-based approaches have emerged in the geospatial science literature as well with researchers geolocating individuals based on the content of their social contributions [Cheng et al, 2013; Hecht et al, 2011; Li et al, 2008] and building location recommendation systems [Bao et al, 2012; Matyas and Schlieder, 2009; McKenzie et al, 2013a], to name a few. It has been shown in previous work that individual words and topics in place descriptions are indicative of geospatial location [Adams and Janowicz, 2012]. However, this effect only becomes present at a coarser spatial resolution; e.g., when comparing meso-level features like cities and national parks.

Methodology

In this section we describe the four attribute types available for POI conflation. Additionally, the properties of the two datasets used for training and evaluation will be described. We continue by presenting a number of attribute-specific conflation methods as well as the metrics and measures applicable to each attribute type. Finally, we propose three distinct, weighted models that combine multiple attributes in order to produce high accuracy venue matching.

Venues Dataset

A random sample of 200 POI were collected from the continental United States through the public Yelp API. Selected businesses were required to have a name, geographic coordinates, at least one category tag, and a minimum of five user-contributed reviews. The 200 POI were manually compared to venues accessed through the Foursquare API returning a positive matched set of 140 Foursquare venues. Again, a positive match required that the above attributes also be present in the matched Foursquare POI. In place of reviews, this other source of user-generated content includes *Tips*, which are similar in nature to *Reviews* except shorter in length, mostly recommending items or offering advice. For a set of POI to be chosen as a match, a minimum of 3 tips associated with the Foursquare venue were required. We call this matched set of POI, V_M. Descriptive statistics for these locations include a combined mean of 26,966 characters (SD = 63,133) and 509 characters (SD = 581) for the Yelp reviews and Foursquare tips respectively.

The mean great circle distance between the two venue sources equated to 62.8 meters. The largest discrepancy in distance between two matched venues in V_M was found to be 869.3 meters. Using this distance as a rough upper-bound, each known Yelp business was buffered to return all Foursquare venues within a 1000 meter radius. This resulted in a test set, V_T, of 73,304 POI averaging 505 per known Yelp location (min=16, max=2770). As shown in Figure 2, all POI in V_T were comprised of some textual name attribute and geographic coordinates, 82.1% of POI were tagged with a minimum of one category and 34.2% listed at least one user-contributed *tip* with a mean of 10.3 tips per venue.

Venue properties & Measures

The methods used to match POI are grouped by the attribute of the venue used as input. These measures focus on the *Name*, *Category*, *Geographic Location*, and *Unstructured descriptive text* assigned to each POI. To start, each attribute is assessed independently with the purpose of determining the independent accuracy of each matching method. Every identified property of each Yelp ID in V_M is matched against all Foursquare venues within a 1000 m radius and ranked by "most-likely match." The Foursquare venue ranked as the "best" match is stored in the first position, second in the second position, etc. The *actual venue match* is then extracted from this ranked list of venues and the ranked position of the venue is stored as a measure of model accuracy.

Venue Name

Levenshtein Distance

Given that both *Yelp* and *Foursquare* necessitate the creation of POI (as well as all attributes associated with these venues) through their application user-base, an initial match-assessment measure could start with the *Name* attribute assigned to a given venue. By far the most common method for measuring the similarity between two sequences of text is the *Levenshtein distance* [Levenshtein, 1966] which is an edit-distance function that assigns a unit of cost to all edit operations. Simply put, the Levenshtein distance between two strings is calculated as the minimum number of edits required to transform one string to another. In this case, edit operations are defined as addition, deletion and substitution with each operation given a weight of 1. The fewer edits needed, the smaller the edit-distance and the more similar the two venue names are to be gauged. The Levenshtein distance between each Yelp business in V_M and the nearest Foursquare venues in V_T are calculated and ranked based on smallest edit distance.

Phonetic Similarity

Another approach to comparing named entities involves considering the phonetics of the POI name. Since POI are contributed by individual users (often via mobile device "on-the-go"), it is not unreasonable to assume that many names may be misspelled, with users focused primarily on how the name "sounds." Initially, two phonetic algorithms, *Soundex* [Russell and Odell, 1918] and *Double Metaphone* [Philips, 2000] were tested to evaluate similarities between names. The Double Metaphone algorithm was chosen as it outperformed the older Soundex approach in early trials. The original Metaphone [Philips, 1990] phonetic algorithm works by encoding the English pronunciation of words as a unique key. Sets of keys are combined as codes which can be used to represent words or phrases based on how they sound. The Double Metaphone approach builds on the earlier implementation, adding the benefit of an alternative phonetic code for each string (often found to be the same as the original). This method attempts to account for alternate pronunciation from various language origins.

Using the Double Metaphone algorithm, two phonetic codes (primary and alternate) are generated for each Yelp business in V_M and two for each of the

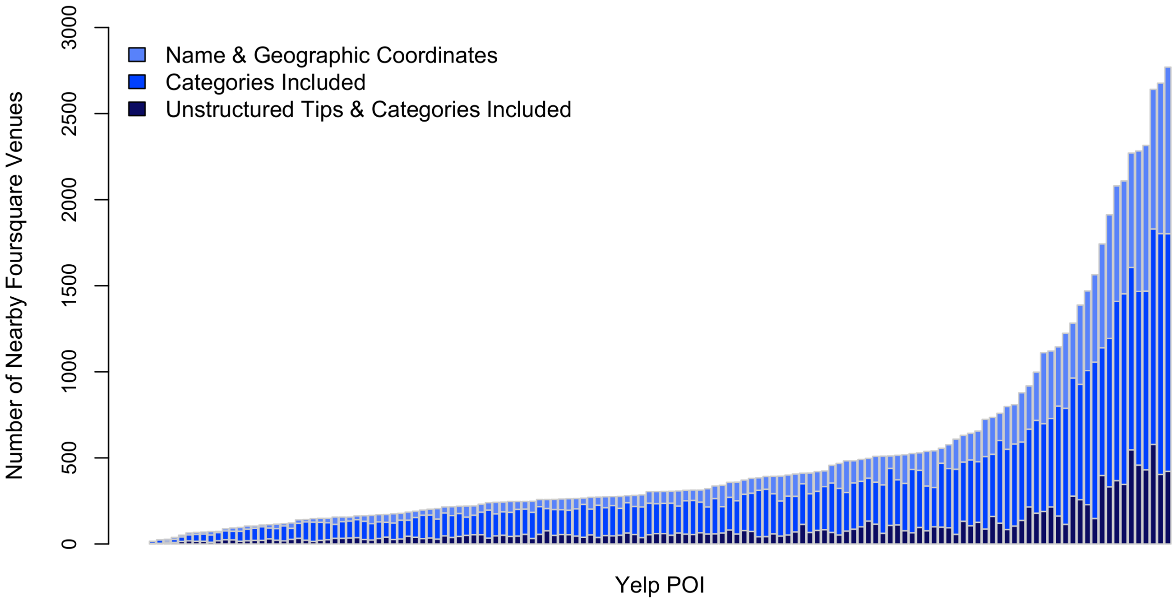


Figure 2: Each Yelp POI shown by Number of Foursquare Venues within 1000m. All POI have *Name* and *Geographic Coordinate* attributes while fewer are tagged with *Categories* and less consist of *Tips*.

venues in V_T. Using the Levenshtein distance metric, each pair of codes is compared, producing four phonetic distance values. As was the case with the previous metric, venues are ranked by distance value from smallest to largest, the smallest value indicating the best estimated match given this similarity measurement method.

Categories

Both Foursquare and Yelp provide a three-tier hierarchical set of category tags from which users can tag a POI. Users can choose to add any number of tags from any tier and across base-tiers. The number and arrangement of categories changes on occasion[2], but as of this writing, Foursquare offers 385 categories and Yelp lists 668 (across all tiers). One option for matching POI is to look at the tagged categories assigned to each venue and measure the similarities between tags.

The first step in exploiting the supplied categories is to trace up the hierarchy tree, assigning parent categories to POI that only contained child-tier tags. The reason for doing this is to increase the probability of matching two places based on category. For example, a venue tagged as "Park" in both sets could be ambiguous, but tracing up the hierarchy tree we find similar, and more descriptive, base classes of “Outdoors & Recreation” and “Active Life.” The inclusion of these simply serves to strengthen the similarity (or dissimilarity) of points based on category.

The second step involves aligning the two source hierarchies. The *Alignment API 4.0* [David et al, 2011] is used to align the sets based on the *Java Wordnet Library*[3]. This results in equivalency class probability values for each pair of categories, based on links within the *Wordnet* corpus. The values are bound between 0 and 1, with 0 indicating a lack of equivalence and 1 indicating complete equality. Provided the set of alignment measures, the categories from each location in V_M are compared to each category tag in each nearby venue in V_T, producing an array of paired measurement values. These measures are averaged for each set of venues and ranked based on value.

Geographic Location

In POI matching, there is an assumption that the geographic distance between two locations is a strong indicator of match accuracy. Though this is highly dependent on the contributing source of the data, in this case, the location of each venue offered by both POI sources is subject to the same contribution errors present in any of the other attributes. Many users either enter an address or cross-street for a venue (which is then geocoded) or, more likely, they rely on the geographic positioning method employed by their mobile device. Given the uncertainty of mobile positioning systems and systematic errors inherent to GPS

[2] http://about.foursquare.com/foursquare-categories

[3] http://sourceforge.net/apps/mediawiki/jwordnet, http://wordnet.princeton.edu/

and wireless positioning, it is not uncommon to find a significant discrepancy in the geographic coordinates of the same location sourced from two applications. As shown in Figure 3, the mean distance between two POI in our matched set V_M is 62.8 meters with a maximum difference of 869.3 meters.

In order to test the theory that geographic distance between venues is a good attribute on which to match venues, we calculate the great circle distances between all locations in V_M to all nearby locations within 1000 meters.[4] We then rank venues purely based on this geographic distance, with the smallest distance representing the best estimated match through this method.

Topic Similarity

Topic Modeling: Descriptive Reviews

Being that a fundamental aspect of both Yelp and Foursquare is to provide descriptive reviews of businesses [Yelp/.com, 2013], it follows that the textual descriptions contributed to the same place in either application would be related in subject matter. In theory, one would expect a greater similarity in the topics mentioned at matched POI than topics discussed at unmatched ones. Building on this theory, an unsupervised topic model approach is taken to measure the similarities across locations in order to determine a match. *Latent Dirichlet allocation (LDA)* is an unsupervised, generative probabilistic model used to infer the latent topics in a textual corpus [Blei et al, 2003]. Here we train LDA by treating the text associated with each venue as a single document. LDA takes a "bag-of-words" approach, "discovering" topics that are represented as multinomial distributions over words. The words that compose the topics emerge from the training set of documents based on co-occurrences of words within and across documents. After training LDA, not only do we have a set of topics but also each document (i.e., POI) is modeled as a probability distribution (or histogram) over these topics.

As input to the LDA model, all Yelp reviews in V_M are merged based on venue and stripped of all non-alpha characters. Two implementations of the model were executed with 40 and 100 topics with the 100 topic model producing the best matched results. After removing venues with less than 40 characters, the percentage of POI on which topic modeling can performed is reduced to 26.1% of V_T. The MALLET toolkit [McCallum, 2002] provided the LDA implementation used in this work.

Once the V_M and V_T POI are represented as topic distributions, the *Jensen-Shannon divergence (JSD)* (Equation 2) is employed to compute a dissimilarity value between two places. V_{Yelp} and V_{FS} represent the topic signatures for a Yelp business and Foursquare venue respectively, $M = \frac{1}{2}(V_{Yelp} + V_{FS})$ and $KLD(V_{Yelp} \parallel M)$ and $KLD(V_{FS} \parallel M)$ are *Kullback-Leibler divergences* as shown in Equation 3.

$$JSD(V_{Yelp} \parallel V_{FS}) = \frac{1}{2}KLD(V_{Yelp} \parallel M) + \frac{1}{2}KLD(V_{FS} \parallel M) \quad (2)$$

$$KLD(P \parallel Q) = \sum_i P(i) \log_2 \frac{P(i)}{Q(i)} \quad (3)$$

The *JSD metric* is calculated by taking the square root of the value resulting from the divergence. Given the inclusion of the logarithm base 2, the resulting metric is bound between 0 and 1 with 0 indicating that the two venue topic distributions are identical and 1 representing complete dissimilarity. Computing the dissimilarity between each known Yelp business and its nearby Foursquare venues in V_T, produces a ranked set of POI from which a match can be extracted.

Topic Modeling: Category Constrained

Provided categorical tags for each venue, a modified version of LDA can be applied that makes use of the user-generated tags. *Labeled Latent Dirichlet allocation (LLDA)* [Ramage et al, 2009] constrains the topic generation process by the tags or categories assigned to each document. The result is a one-to-one relation between each tag and a LDA topic. Since the majority of POI are associated with multiple categories, the multi-label approach inherent to LLDA generates latent topics that directly correspond to categories. As is the case with LDA, each location consists of a distribution across topics, though in this case, the assignment of words to those topics is restricted by the categorical label of the POI. The model was executed through the use of the Stanford Topic Modeling Toolbox.[5]

Mirroring the LDA method, each point can be described as a distribution across topics, though in this case, the number of topics is set to 349, the number of category tags present in V_M. Again, the Jensen-Shannon divergence metric was used to rank POI distributions in order of dissimilarity.

Weighted Multi-attribute Model

The above methods for matching POI are based on single attributes. Each method has its strengths and weaknesses and some produce a higher percentage of correct matches than others (as shown in section 4). In order to truly return the best possible match, the previously mentioned matching measures must be blended to produce a model that is better than simply the sum of its parts. One potential option is to merely distribute the modeling weight evenly across all attributes. While this method does make

[4] The PostGIS function *ST_Distance* is used with the Geography datatype. http://postgis.refractions.net/docs/ST_Distance.html

[5] http://nlp.stanford.edu/software/tmt/tmt-0.4/

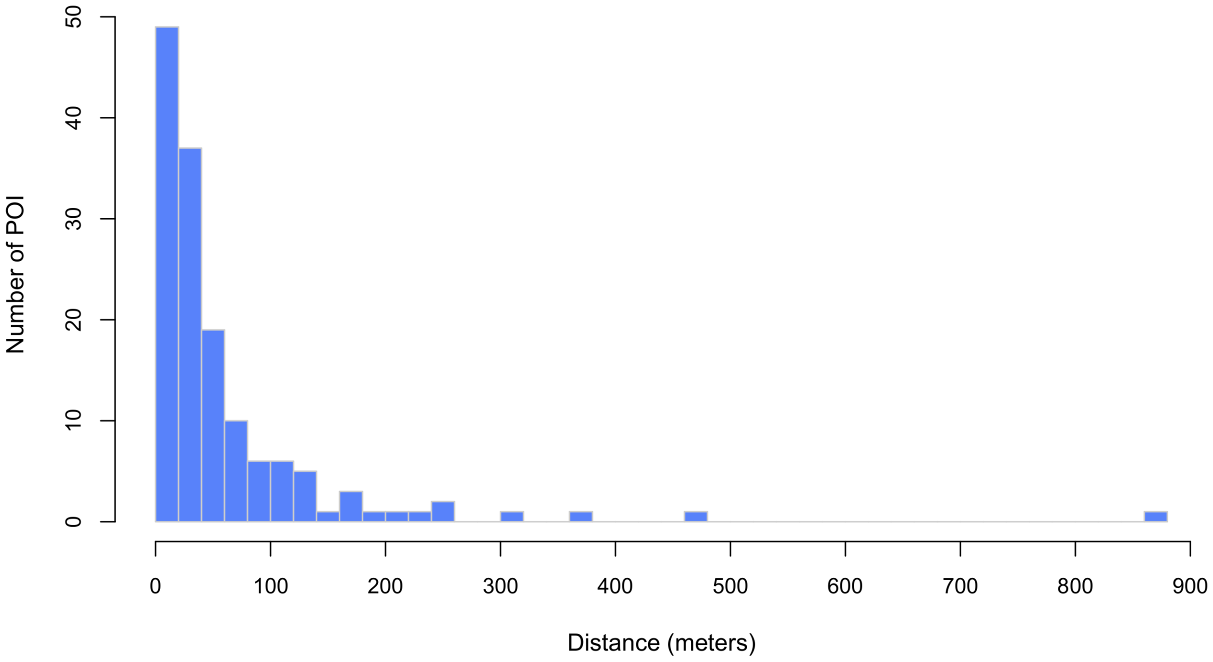


Figure 3: Histogram of distances between matched venues.

use of all attributes, it does not take in to account the performance of each model in comparison to others.

In taking a more practical approach, we first look at the results of each attribute-based method individually and rank them in terms of match accuracy. The application of weights can then be distributed based on the ordinal performance ranking of each method. Equation 4 shows how the weights are calculated, where N is the total number of methods and M_j is the ranked performance of each model. For example, assuming $N = 5$, the weights assigned to the top two ranked measures, W_1 & W_2, would be 1/3 and 4/15 respectively.

$$W_j = \frac{N - M_j + 1}{\sum_{j=1}^{n} N - M_j + 1} \tag{4}$$

A downside of this approach is that it does not take in to consideration the true *measured* performance of each model. In order to build a weighted model based on measured outcomes, a binomial probit regression model is used to estimate the overall contribution each attribute makes in correctly determining a match. The positioned rank for each distinct attribute, measured across all 140 venues are entered as independent variables to the model. The dependent "correct match" variable consists of either a 1 (match) or 0 (no match) for each pair of venues. The coefficients resulting from the model are normalized and applied as weights to a *regression-based weighted multi-attribute model*. Depending on the number and combination of attributes being considered, the regression-based weights will shift.

Evaluation

In this section, the results are presented independently for each attribute method and compared with the results of the two *attribute-weighted* models and a baseline unweighted model. Our approach ranks each Foursquare POI in V_T by its attribute-matching measure (e.g., Levenshtein) to the associated Yelp POI in V_M (discussed in section 3.2). Proceeding through the ranked list of items, the position of the actual attribute match is recorded. A perfect match would result in the correct Foursquare location matching to the top ranked POI in the attribute ranked set. A second position rank indicates that the attribute model chose the "correct Foursquare POI" in V_M as its second most likely match, and so on. Table 1 shows the match percentage for each ranked position of each independent measure.

The ranked percentages for *Categories*, *LDA*, and *LLDA* are based on subsets of the data as not all POI were tagged with categories or associated with unstructured content. For example, a randomly selected point in V_M is queried against 7222 possible matches in V_T based on name and distance matching methods, 5775 POI for category matching, and 1372 POI for unstructured text-based models. While the reduced number of POI demonstrates no effect on our results, shown in Table 1, it implies that a model built on all possible POI has potential to perform much worse

Position	Levenshtein	D Metaphone	Distance	LDA	LLDA	Categories
1	86.5	85.8	51.8	61.7	39.0	23.4
2	3.5	4.3	13.5	14.2	13.5	12.8
3	2.1	2.1	11.3	5.0	12.8	5.0
4	1.4	0.0	4.3	2.8	2.1	7.8
5	0.7	0.0	5.0	2.1	3.5	4.3
6	0.0	1.4	2.8	1.4	2.8	3.5
7	0.0	0.0	1.4	0.0	2.1	3.5
8	0.0	0.0	2.1	1.4	2.1	1.4
9	0.0	0.0	0.7	2.1	0.7	2.8
10	0.0	0.0	0.7	0.7	0.7	2.1

Table 1: Independent attribute method by percentage of top ten ranked positions. Note that both the Categories, LDA, and LLDA results are based on subsets of the venues.

than one based on a restricted subset (e.g., a rank of 7221 out of 7222 venues is considerably worse than a rank of 1371 out of 1372 venues). This difference in POI counts may have a significant impact should our independent models produce results completely at random or if they were based on an extremely small subsets (e.g., 2 POI). The fact that we see no difference in our rank percentages when restricting all possible matches to the smallest subset (LDA) suggests that this restriction has a negligible effect on the results of the models, if any.

The Name Attribute

Overall, the *Levenshtein Distance* measure performed the best of the independent methods. In fact, 90.0% of the venues on V_M were matched within the first two ranked positions, achieving an accuracy measure considerably better than the 80% we were aiming for. Given these results it is not surprising that the *Levenshtein Distance* is one of the most common methods of name matching Peng et al [2012].

The *Double Metaphone* approach was slightly less effective at predicting matched venues in the first position, though relaxing the match criteria to two positions resulted in an estimation accuracy of 90.1%, slightly higher than that of the *Levenshtein Distance*. While the two methods produced strong match results, measuring the congruency (Equation 5) of the results returned a cosine value of 0.491, suggested that the two measures explained a lot of the same matches.

$$M_c = \frac{\sum V_{Lev} V_{DMeta}}{\sqrt{\sum V_{Lev}^2 \sum V_{DMeta}^2}} \quad (5)$$

Categories

The *Category Alignment* method performed the poorest of all methods with only 23.4% of POI being correctly matched. The positive estimation percentage does not reach 50% until the fifth ranked position is included, indicating that there is a substantial amount of uncertainty associated with a match estimation based purely on categorical alignment.

Geographic Distance

Of the two measures built on required properties of both POI sources (i.e., name and geographic coordinates), matching based on distance produced the lowest match percentages. Only 51.8% of POI were correctly matched on the first try using the great circle distance measure. One would need to relax position selection to the fifth most likely match in order to approximate the top performing attribute measure (Levenshtein). These results contradict the widely made assumption that proximity of POI is a strong match indicator (i.e., that "geometry trumps semantics" when performing conflation tasks [Adams et al, 2010]). As mentioned in section 3.5, the user-generated aspect of this data is most likely to blame for this level of inaccuracy. Georeferencing [Goldberg, 2013] the POI in V_M shows that Foursquare venues are displaced by a mean of 49.0 meters and Yelp businesses by 37.1. While Yelp is slightly more accurate than Foursquare, these two user-contributed datasets seem to share similar issues. Restricted to a single independent attribute on which to match POI, these results suggest that one would be much better off using the *Name* attribute.

Topic Similarity

The introduction of descriptive, textual review data offered the ability to match venues on a characteristic unique to this form of user-generated content. While not all venues in the test set consisted of review and tip data, the results based on this limited subset of venues were quite encouraging. Based purely on publicly available, unstructured text contributed from a wide range of users, the same POI could be correctly matched across two providers, 61.7% of the time. These results indicate that there is a commonality that exists between data providers. Topics related to a venue in one dataset correspond to topics associated with that exact same venue in another dataset. Using a model built from 100 topics, 75% of POI can be matched within two ranked match positions.

Constraining the topic generation process by categories associated with a venue (Labeled LDA) proved to be less fruitful than the unconstrained model. Only the equivalency class category alignment method produced inferior results, suggesting the abandonment of the labeled model in favor of the standard LDA method. This is not to say that the labeled model did not produce useful results, in fact they were markedly better than one would expect at random, but not nearly as effective as the unrestricted topic model.

Weighted Multi-Attribute Models

Though each independent matching method exhibited excellent results, models that merged these methods proved far superior. Both the number of independent models and weight applied to each model was varied in order to produce a model which resulted in the highest achievable match accuracy. In view of the fact that not all POI are associated with review data, two categories of the weighted-models were constructed: a *Three-Method* model and a *Four-Method* model. Since match-accuracy based on the categories attribute was so low, a third model was not necessary as the inclusion of this data did not increase match accuracy.

Three-Method Models

The first category of model utilized a combination of the three independent attribute measures that can be applied to all venues. As shown in Table 2, irrespective of the way in which the methods were merged, these multi-attribute models outperformed each of the independent match methods.

Position	Unweighted	Ordinal	Regression
1	87.2	87.2	87.2
2	6.4	5.7	7.1
3	1.4	2.1	1.4
4	0.0	0.0	0.7
5	0.0	0.0	0.7
6	1.4	0.7	1.4
7	0.0	0.0	0.0
8	0.0	0.0	0.0
9	0.0	0.0	0.0
10	0.0	0.7	0.0

Table 2: Weighted Three-Method models by percentage of top ten ranked positions. These models is based on all venues in set V_T.

As a baseline, an unweighted model was constructed that averages the influence of the three independent methods. This unweighted method was quite accurate, correctly matching 87.2% of POI. The *Ordinally weighted* model showed no increase in accuracy for first position ranked matches and actually displayed a decrease in second position matches when compared to the unweighted model. This decrease in accuracy is mirrored in the *Four-Method* models shown in Table 3. The reason for this decrease in accuracy can be explained by examining the number of attributes that are being exploited. This *Three-Method* model applies two independent methods to one of two attributes. Since these two methods based on the name attribute outperform the distance-based method, they are assigned weights of 0.5 and 0.3 based on Equation 4. This means that 80% of the *Ordinally weighted Three-Method* model is founded on the *Name* attribute and only 20% comes from the *Distance* attribute.

The third of these models takes a regression-based approach to assigning weights. The results of the binomial logit regression model produce the *Regression-based Three-Method* model shown in Equation 6 where *Lev*, *DM* and *Dist* represent the *Levenshtein*, *Double Metaphone* and *Distance* independent measures respectively. This regression-based model M_{reg}, while matching the other two models in percentage of first position based matches, did pull ahead when incorporating second position POI.

$$M_{reg} = 0.680Lev + 0.112DM + 0.208Dist \tag{6}$$

Four-Method Models

Reducing the number of POI on which the independent methods were evaluated allowed us to incorporated reviews and tips in the form of unstructured text. The addition of this attribute proved to have a significant impact on the accuracy of POI matching. Comparatively, the results shown in Table 3 reflect those of the *Three-Method* models, with the notable exception of a substantial increase in accuracy.

Position	Unweighted	Ordinal	Regression
1	95.7	94.3	97.9
2	1.4	1.4	0.7
3	0.0	0.7	0.7
4	1.4	0.0	0.0
5	0.0	0.7	0.0
6	0.0	0.0	0.0
7	0.7	0.7	0.0
8	0.0	0.7	0.0
9	0.0	0.0	0.0
10	0.0	0.0	0.0

Table 3: Weighted Four-Method models by percentage of top ten ranked positions. These models are based on a subset of venues in set V_T.

Again, all three versions of the model outperformed each attribute-specific method independently. Both the unweighted and ordinally weighted models performed very well with correctly matched results landing in the 95% region. The top performing model was based on a regression of the four independent attribute methods, producing remarkable match-accuracy of almost 98% correct matches across

140 POI. The regression-based weights are shown in Equation 7.

$$M_{reg} = 0.562Lev + 0.094DM + 0.170Dist + 0.174LDA \quad (7)$$

Evaluating the Model

The models constructed in the sections above were based on a random selection of 140 of manually matched POI from the Foursquare and Yelp datasets. The top performing multi-attribute model was constructed based on the results of a regression model built on these 140 POI. In order to test the validity of this model, we randomly selected 100 new POI in the Yelp dataset that were manually matched to the same number of POI in the Foursquare venue set. These test POI were evaluated by the four top performing independent methods as well as the regression-based model proposed in section 4.5. The evaluation results are shown in Table 4.

The four independent measures illustrate results similar to those seen in Table 1 with both *name matching* methods producing high levels of accuracy and the *Distance* and *LDA* attributes showing comparable accuracy percentages. The match-accuracy of the *Regression-weighted* model correctly matched **97%** of the 100 POI sample, validating the technique and proposed model.

Matcher Scores and Identity

So far, we only considered cases where a *true positive* matching POI exists between the datasets. Despite their overall fair quality and coverage, however, many Foursquare and Yelp POI lack multiple attributes or have no matching counter part. This becomes even more apparent as we add more data sources such as OpenStreetMap, WikiMapia, Google Places, Yahoo Local, GeoNames, and so forth. Ideally, one would just run the presented weighted multi-attribute matcher as a RESTful Web Service to match POI on-the-fly. This will create a growing dataset that does not contain the POI data as such[6] but instead lists *identity assumptions*. The power of this idea is well recognized and at the very core of Linked Data [Bizer et al, 2009]. Following the Recourse Description Framework (RDF) and the Web Ontology Language (OWL), each match can be defined as a triple

$$< poi - uri > owl : sameAs < poi' - uri > .$$

The owl:sameAs relation establishes identity between individual resources and consequently is a transitive property; see [Hitzler et al, 2011] for the formal semantics of OWL (and RDF). To give a concrete example, the French Press Cafe used in the introduction would be represented by the following triples (in N3 notation):

```
<http://4sq.com/8dLBOO>
  owl:sameAs <[...]yelp.com/biz/the-french-press-santa-barbara>;
  owl:sameAs <[...]plus.google.com/105818558285682895797/>;
  owl:sameAs ...
  .
```

Consequently, data about these places could be linked and conflated on-the-fly. In fact, the authors are currently establishing such a database/triplestore. This, however, requires a match score threshold from which on POIs are considered equal (owl:sameAs). Furthermore, this score has to be robust in a sense that some POI data will not contain tips or categories, and so forth.

While our approach provides *true positive* matches in 97% of all cases given there is such a match, one would expect that this success rate drops significantly as soon as we relax this condition and also add new data sources. We selected 200 random Foursquare POI and performed an automatic matching to other data sources without ensuring that there is, in fact, a *true positive* match in these sources. Consequently, this sample contains POI that cannot be matches as well as those that miss certain attributes (in Foursquare as well as the other sources such as Yelp). We recorded the match score to the best proposed match and evaluated all matches manually. The resulting mean match score for ***true positives* is 9.52**. More interesting, however, is the rate of *false positives*. In our random sample a match score value of **26.4** will find 95% correct matches but also return 65% false matches. A value of **10.04** will return only 5% false matches but only find 47% correct matches. This leads to an F-score of 0.32 and 0.35, respectively. These numbers should be interpreted with caution. By always selecting the best match, we force the matcher to chose *false positives* and also to compute matches in the presence of missing attributes. For example, a matched-to POI may have no tips and no categories assigned to it. These are stricter conditions than typically applied to F-scores (necessarily leading to weaker results). However, they allow us to discover nearby similar POI automatically.

In other terms, we will require additional predicates that can formally express a weaker notion of equivalence. Such matching relations, e.g., skos:closeMatch have been propose by the SKOS W3C Recommendation[7] and their formal characterization is part of ongoing research [Halpin et al, 2010]. We leave this part to future work.

Conclusions and Outlook

In this work we addressed the problem of matching Points of Interest from the location-based social network Foursquare and the local directory service Yelp,

[6] this would not be in line with the usage regulations of some of the data providers.

[7] See http://www.w3.org/TR/skos-reference/#mapping.

Position	Levenshtein	D Metaphone	Distance	LDA	Regression-based
1	76.0	87.0	51.0	63.0	97.0
2	5.0	7.0	16.0	14.0	1.0
3	3.0	0.0	11.0	6.0	0.0
4	0.0	1.0	4.0	2.0	1.0
5	2.0	0.0	5.0	2.0	0.0
6	2.0	0.0	1.0	1.0	1.0
7	1.0	0.0	0.0	1.0	0.0
8	0.0	0.0	0.0	2.0	0.0
9	1.0	1.0	2.0	0.0	0.0
10	0.0	0.0	4.0	2.0	0.0

Table 4: The top 4 independent attribute matchers by percentage match with the top performing weighted multi-attribute model.

both being widely recognized as market leaders in terms of innovation and market share. While their datasets overlap, they focus on different aspects of places. Foursquare specializes on social, *placial* interactions and records user check-ins, activities, and short tips (comments). In contrast, Yelp is specialized on providing detailed user reviews and an increasing range of semi-structural place attributes such as noise level, price range, ambiance, wifi availability, and so forth. However, the temporal dimension present in Foursquare data is missing entirely from Yelp. Being a social online network, Foursquare offers a broad range of liberal APIs to access their data, while the interfaces provided by Yelp are rather minimal and focused on restricting access to their data. This makes matching data from both services difficult.

Nonetheless, conflating their datasets is very attractive from a research perspective. There is sufficient overlap between the attributes stored by Yelp and Foursquare to support matching, and enough differences to justify the effort of conflation. The presented work focuses on the first step of POI conflation, namely identifying whether two information entities refer to the same place in the physical world. In the presented work, we demonstrated a weighted multi-attribute matching strategy that can successfully match **97%** of randomly selected Yelp POI to their corresponding Foursquare entities. We also investigated what match score is suitable for the automatic on-the-fly matching of POI from different sources and in the presence of missing attribute values. Our matching strategy is developed in a fashion that does not violate the API policies and data is only stored for caching purposes. Being able to match and later conflate POI will enable researchers to better explore human mobility, consumer behavior, and social interaction.

As the number of volunteered geographic information stores increase, so have the attributes associated with the data. Names and geographic coordinates persist, while new attributes such as activity categories and descriptive text have appeared. We are no longer reliant on name matching, place types, and geographic proximity. In fact, our approach has shown that the distance between matched POI from different providers can be substantial and matching points based on geographic location alone is often imprudent. In the results section we touched on some of the reasons why there may be discrepancy in user-generated locations and how this discrepancy varies across providers.

Future work in this area will involve enhancing our models to match additional user-generated and non-user-generated POI datasets. There are an enormous amount of geographically referenced data publicly available online and identifying the same POI in different datasets is a substantial step forward in the aspiration of POI data conflation. Specifically, in the future we will test our work against less controlled VGI datasets such as OpenStreetMap or Wikimapia. We assume that matching based on geographic distance and common name will be less effective and that the use of categorization will be less homogeneously applied throughout the dataset. While this paper makes use of the more prominent properties of POI, additional attributes can and should be exploited. For example, semi-structured price-range values for POI can be compared as well as user-contributed *star* rankings. Additional features of the data such as sheer number of reviews or tips can be employed, as well as the home locations of the contributors. Finally, in the future we also plan to look into the second step of conflation, namely how to merge attribute *values*.

References

Adams B, Janowicz K (2012) On the geo-indicativeness of non-georeferenced text. In: Breslin JG, Ellison NB, Shanahan JG, Tufekci Z (eds) The Internatioal AAAI Conference on Weblogs and Social Media, The AAAI Press, pp 375–378

Adams B, Li L, Raubal M, Goodchild MF (2010) A general framework for conflation. Sixth International

Conference on Geographic Information Science, Extended Abstract

Bao J, Zheng Y, Mokbel MF (2012) Location-based and preference-aware recommendation using sparse geo-social networking data. In: 20th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Redondo Beach, CA, USA

Bilenko M, Mooney RJ (2003) Adaptive duplicate detection using learnable string similarity measures. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp 39–48

Bizer C, Heath T, Berners-Lee T (2009) Linked data – the story so far. International Journal on Semantic Web and Information Systems 4(2):1–22

Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. Journal of Machine Learning Research 3:993–1022

Chen CC, Knoblock CA, Shahabi C (2006) Automatically conflating road vector data with orthoimagery. GeoInformatica 10(4):495–530

Cheng Z, Caverlee J, Lee K (2013) A content-driven framework for geolocating microblog users. ACM Transactions on Intelligent Systems and Technology 4(1):2

Cohen WW, Ravikumar P, Fienberg SE, et al (2003) A comparison of string distance metrics for name-matching tasks. In: Proceedings of the IJCAI-2003 Workshop on Information Integration on the Web, pp 73–78

David J, Euzenat J, Scharffe F, Trojahn dos Santos C (2011) The alignment api 4.0. Semantic Web 2(1):3–10

Devogele T (2002) A new merging process for data integration based on the discrete frechet distance. In: 10th international symposium on spatial data handling, vol 2, pp 167–181

Elmagarmid AK, Ipeirotis PG, Verykios VS (2007) Duplicate record detection: A survey. Knowledge and Data Engineering, IEEE Transactions on 19(1):1–16

Freeman AT, Condon SL, Ackerman CM (2006) Cross linguistic name matching in english and arabic: a one to many mapping extension of the levenshtein edit distance algorithm. In: Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp 471–478

Goldberg D (2013) Texas a&m university geoservices. http://geoservices.tamu.edu, accessed: 03/05/2013

Halpin H, Hayes PJ, McCusker JP, McGuinness DL, Thompson HS (2010) When owl: sameas isn't the same: An analysis of identity in linked data. In: The International Semantic Web Conference, Springer, pp 305–320

Hastings J (2008) Automated conflation of digital gazetteer data. International Journal of Geographical Information Science 22(10):1109–1127

Haunert JH (2005) Link based conflation of geographic datasets. In: 8th ICA workshop on generalisation and multiple representation, A Coruña, Spain

Hecht B, Hong L, Suh B, Chi EH (2011) Tweets from justin bieber's heart: the dynamics of the location field in user profiles. In: Proceedings of the 2011 annual conference on Human factors in computing systems, ACM, pp 237–246

Hitzler P, Krötzsch M, Rudolph S (2011) Foundations of semantic web technologies. Chapman and Hall/CRC

Joachims T (1998) Text categorization with support vector machines: Learning with many relevant features. European Conference on Machine Learning pp 137–142

Jones CB, Purves RS (2008) Geographical information retrieval. International Journal of Geographical Information Science 22(3):219–228

Lait A, Randell B (1996) An assessment of name matching algorithms. Technical Report Series-University of Newcastle Upon Tyne Computing Science

Lamburt L, Koyfman L, Ponte J (2002) Data merging techniques. US Patent 6,374,241

Levenshtein VI (1966) Binary codes capable of correcting deletions, insertions. Soviet Phys Dokl 10:707–710

Li L, Goodchild MF (2011) An optimisation model for linear feature matching in geographical data conflation. International Journal of Image and Data Fusion 2(4):309–328

Li Q, Zheng Y, Xie X, Chen Y, Liu W, Ma WY (2008) Mining user similarity based on location history. In: 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM, Irvine, CA, USA, p 34

Matyas C, Schlieder C (2009) A spatial user similarity measure for geographic recommender systems. GeoSpatial Semantics pp 122–139

McCallum AK (2002) Mallet: A machine learning for language toolkit, http://mallet.cs.umass.edu

McKenzie G, Adams B, Janowicz K (2013a) A thematic approach to user similarity built on geosocial check-ins. In: The 15th AGILE International Conference on Geographic Information Science, Leuven, Belgium

McKenzie G, Janowicz K, Adams B (2013b) Weighted multi-attribute matching of user-generated points of interest. In: 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems - Short Paper, Orlando, FL, USA

Mülligann C, Janowicz K, Ye M, Lee WC (2011) Analyzing the spatial-semantic interaction of points of interest in volunteered geographic information. Spatial Information Theory pp 350–370

Page L (2001) Method for node ranking in a linked database. US Patent 6,285,999

Peng T, Li L, Kennedy J (2012) A comparison of techniques for name matching. International Journal on Computing 2(1)

Philips L (1990) Hanging on the metaphone. Computer Language 7(12 (December))

Philips L (2000) The double metaphone search algorithm. C/C++ Users Journal 18(6):38–43

Ramage D, Hall D, Nallapati R, Manning CD (2009) Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Singapore, pp 248–256, URL http://www.aclweb.org/anthology/D/D09/D09-1026

Reed WJ (2001) The pareto, zipf and other power laws. Economics Letters 74(1):15–19

Russell RC, Odell MK (1918) Soundex. US Patent 1

Scheffler T, Schirru R, Lehmann P (2012) Matching points of interest from different social networking sites. KI 2012: Advances in Artificial Intelligence pp 245–248

Sehgal V, Getoor L, Viechnicki PD (2006) Entity resolution in geospatial data integration. In: Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems, ACM, pp 83–90

Wu Y, Winter S (2009) Inferring relevant gazetteer instances to a placename. In: 10th International Conference on GeoComputation. UNSW, Sydney, Australia

Yelp/com (2013) Yelp: About. https://yelp.com/about/, accessed: 03/03/2013