Toward Traffic-Driven Location-Based Web Search

Zhiyuan Cheng, James Caverlee, Krishna Y. Kamath, and Kyumin Lee
Department of Computer Science and Engineering
Texas A&M University
College Station, TX, USA
(zcheng, caverlee, kykamath, kyumin)@cse.tamu.edu

ABSTRACT

The emergence of location sharing services is rapidly accelerating the convergence of our online and offline activities. In one direction, Foursquare, Google Latitude, Facebook Places, and related services are enriching real-world venues with the social and semantic connections among online users. In analogy to how clickstreams have been successfully incorporated into traditional web ranking based on content and link analysis, we propose to mine traffic patterns revealed through location sharing services to augment traditional location-based search. Concretely, we study location-based traffic patterns revealed through location sharing services and find that these traffic patterns can identify semantically related locations. Based on this observation, we propose and evaluate a traffic-driven location clustering algorithm that can group semantically related locations with high confidence. Through experimental study of 12 million locations from Foursquare, we extend this result through supervised location categorization, wherein traffic patterns can be used to accurately predict the semantic category of uncategorized locations. Based on these results, we show how traffic-driven semantic organization of locations may be naturally incorporated into location-based web search.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Application—*Data mining*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; J.4 [**Computer Application**]: Social and Behavioral Sciences

General Terms

Algorithms, Experimentation

Keywords

Location-based search, traffic pattern, checkin, spatio-temporal data mining

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

1. INTRODUCTION

After the explosive growth of on-line social networks like Facebook and Twitter, we are beginning to see a similar rise in location sharing services. These services – like Foursquare, Google Latitude, Facebook Places, among many others – allow users to voluntarily annotate the real world with "checkins" indicating the specific time that a user was at a particular location, often through a smartphone application. In the aggregate, these checkins – along with other user-generated descriptors supported by these services like tags, ratings, and comments – have resulted in billions of explicit "geo-semantic" markers that link people, places, and their activities. As these services continue to grow, there are great opportunities for extremely granular temporal and spatial mining of human mobility, as well as new mobile+location-based services, augmented traffic forecasting, and urban planning.

In this paper, we propose one direction in which location-sharing services may have strong impact – in augmenting traditional location-based web search. Location-based web search (also known as local search) has drawn intensive attention in both industry (Google Maps, Yelp, Yahoo! Local, and Yellow Pages), and the academic community (e.g., [2, 6, 17]). Nearly all the current location-based search systems typically provide rankings for nearby venues based on a user's query and current location. For example, a local search for "coffee" may return a map and an associated ranked list of nearby coffee shops and coffee bean wholesalers. Some of the factors that location-based search engines use for ranking venues in response to a query include: (i) the distance between the user and the target venue; (ii) category analysis of venues (e.g., to group all coffeeshops in a pre-processing step); (iii) the overall ratings for the venue (which are often available for commercial places of business like restaurants); (iv) query and click popularity of the venue's associated web page (e.g., Starbucks may be considered more popular from its web presence than a local coffee shop); (v) reputation of the location via PageRank-style link analysis of the web graph; and (vi) content-based relevance between the query and the location's description (e.g., via information retrieval similarity between the query and a summary of the venue on Yelp or the content from a location's web presence).

In many ways analogous to how clickstreams [11, 26, 5] have been successfully incorporated into traditional search systems based on content similarity [20] and link analysis [12] by connecting real-world user actions (clicks) to relevance, this paper proposes that the *temporal dynamics embedded in the checkins* from location sharing services have

great potential to augment traditional location-based search systems by connecting real-world actions (checkins) to relevance. To illustrate:

- Mike wants to make a reservation for a tennis court on Saturday afternoon so that he can teach his son to play tennis without being disturbed by nearby players. Hence, a local search for "tennis courts" could be augmented with the temporal dynamics mined from location-sharing services to indicate which courts are at off-peak times in terms of player traffic.
- Tina and her friends are going to celebrate their graduation on a Thursday evening and are looking for late-night hot spots. Which local bars are at-peak in terms of traffic? Or will be peaking by the time Tina and her friends arrive?
- John plays a lot of basketball. He usually goes to Williams Park during Wednesday early evening, and Saturday afternoon, which are both free time for him and peak times for other players to get together and play basketball. Suppose John moves to a new neighborhood and wants to find places nearby that have similar traffic patterns, so that he can meet new friends there and play basketball. A traffic-driven location-based search can also easily handle this kind of queries by returning semantically correlated venues with similar traffic patterns.

In all three cases, factors traditionally considered for location-based web search – like distance, overall venue reputation and popularity – are less important than fine-grained temporal dynamics of the traffic patterns of the target venues. Hence, there is an opportunity to augment these traditional approaches with real-world user actions revealed through location sharing services.

In this paper, we propose to study the potential and viability of mining traffic patterns revealed through location sharing services to augment traditional location-based search. As a first step, we propose to model each venue by a *traffic pattern* – essentially a frequency function corresponding to each venue. Two essential and open questions are (i) whether such a model, as compared to traditional content-based and popularity-based models of location-based search, encodes semantically meaningful information; and (ii) whether there is wide enough coverage of location sharing services to support large-scale application of traffic patterns to location-based search. With these questions in mind, this paper makes the following contributions:

- First, we present in this paper a large-scale study of every venue in Foursquare, totaling 12 million unique venues annotated by users of Foursquare via checkins. Based on this study, we propose and evaluate a *traffic pattern*-based model of venues through an investigation of the location-based traffic patterns mined from 22 million checkins from Foursquare and other location sharing services.
- Second, we propose a measure of semantic correlation across venues for organizing venues according to the traffic patterns revealed through location sharing services. Based on this measure, we propose and evaluate a traffic-driven location clustering algorithm that can group semantically related locations with a best-effort performance of F1-Measure 0.675, and Purity 0.764, a critical function for a location-based search engine.
- Third, we observe significant sparsity of checkin data for venues on the "long tail", and so we propose and evaluate a traffic pattern-driven approach for supervised location categorization, wherein traffic patterns can be used to accurately predict the semantic category of uncategorized locations with a F1-Measure of almost 0.8.
- Finally, based on these results, we show how traffic-driven semantic organization of locations may be naturally incorporated into location-based web search through two example scenarios.

2. RELATED WORK

Increasing focus has been put on location sharing services in recent years. Ye et al. [28] proposed friend-based collaborative filtering algorithms to recommend locations utilizing a dataset scraped from Foursquare; Lindqvist et al. [15] analyzed how and why people use location sharing services; Cheng et al. [3] explored human mobility patterns and factors that affect the mobility patterns based on a sampled dataset from location sharing services; and Noulas et al. [18] analyzed user checkin dynamics, and user activities in location sharing services. Compared to these previous studies, this paper focuses on analyzing the temporal traffic patterns revealed from location sharing services and how the traffic patterns can be utilized to enhance traditional location-based search.

Several studies have analyzed the temporal dynamics of on-line social networks and other web corpuses. Golder et al. [10] explored the temporal dynamics associated with on-line social tagging activities. Researchers in [13] studied how queries, their associated documents, and the query intent change over time by analyzing query log data. Temporal evidence was incorporated into models of semantic relatedness for words in [19]. Yang et al. [27] proposed a clustering algorithm that groups temporal patterns associated with online content, and studied how the popularity of the content grows and fades over time. A temporal correlation measure was introduced and applied to study semantic similarity between queries by Chien et al. [4].

Related to our temporal model of traffic patterns, in terms of time series data analysis, Fu [8] provides comprehensive summary on the existing time series data mining literature including representation, indexing, similarity measure, segmentation, visualization and mining. A numerosity reduction component was proposed by Xi et al. [25] and proved to speed up the best performing classifier of one-nearest-neighbor with Dynamic Time Warping (DTW). [14] provided a survey for techniques in time-series data clustering, and corresponding evaluation metrics.

Location-based web search has drawn intensive attention in both industry (Google Maps, Yelp, Yahoo! Local, and Yellow Pages), and the academic community. Early research efforts (e.g., [2, 6, 17]) mainly focused on the extraction of geographic information from page content and structure. Several studies [9, 1, 21] showed that more than one fifth of the queries in general web search systems were geographical relevant queries. Geotagging and gazeteers have been widely used in [16, 22, 23] to augment location-based web search. Watters and Amoudi [24] proposed a method to assign location coordinates to URLs, and a corresponding framework for location-based ranking of search results.

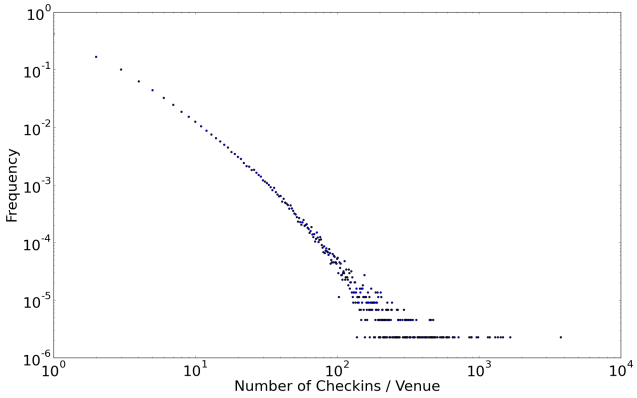


Figure 1: Distribution of # of Checkins Per Venue

3. LOCATION SHARING SERVICES

In this section, we introduce the location sharing service data, present our sampling strategy, and provide a characterization of the venue data collected from Foursquare.

3.1 Sampling Checkin Data

To start with, firstly we need a set of checkins. While Foursquare, Facebook Places, and other related location sharing services are rich resources, all restrict access to a user's immediate social circle and hence are unavailable for public sampling. Hence, we adopt a data collection technique that relies on sampling location sharing status updates from the public Twitter feed. While users of location sharing services in general and the subset who choose to advertise their location via Twitter may not be a representative sample, these status updates are inherently public (mitigating concerns over privacy violations that would arise from mining services like Facebook Places) and offer a rich vein of checkin data. Specifically, we monitor Twitter's public streaming API and search API from October 2010 to January 2011, and collected a set of more than 22 million checkins. Worth mentioning, our data is available on-line at link.

Each checkin contains a fine-granularity location (latitude and longitude) and a timestamp. More than 62% (∼14 million) of the checkins are associated with a venue, and in total 603,796 venues are referenced. Note that since each venue has on average only ∼23 checkins (with a skewed distribution, where some venues are heavily "checked in" to, but the majority have only a handful of checkins as plotted in Figure 1), we aggregate all checkins for venues based on the venue name (e.g., grouping all instances of "Starbucks") for the analysis in the rest of the paper. Venues in the set may be associated with varying degrees of spatial granularity based on the bounding box linked to the venue – from country to province / state to city to district and finally to points-of-interest. In this paper, we mainly focus on the checkins corresponding with the 515,862 point-of-interest venues, each of which is finely geo-labeled with a latitude and longitude.

3.2 Crawling Foursquare Venues

Each venue posted to Twitter has a corresponding "venue page" hosted by Foursquare. To retrieve more information about the venues, we crawled the entire Foursquare-sphere, resulting in nearly 20 million "venue pages" in HTML format. Based on our best-effort parser, we successfully parse

Table 1: Distribution of Venue Categories

Category	Percentage	# of Venues
Home, Work and Other	31.7%	2,457,172
Food	24.3%	1,886,875
Shop	17.1%	1,329,185
Travel Spot	7.0%	541,482
Great Outdoors	6.4%	493,635
Nightlife Spot	5.7%	438,400
Arts & Entertainment	4.3%	334,700
College & University	3.5%	271,825

12,677,314 html pages of venues. Specifically, each venue is stored as the tuple *venue(venueID)* = {*venueID, name, latitude, longitude, address, city, region, postal_code, categories, tags*} An example tuple of a venue is: *venue(877)* = {*877, "once upon a tart", 40.7267, -74.0019, "135 sullivan st", "new york", "ny", 10012, "sandwiches, salad, bakery", "salads, roast pork sandwich, strawberry lemonade ginger iced tea, tarts, desserts"*}.

3.2.1 Venue Characterization

Among the 12 million venues, 56.4% (i.e., 7,147,755) of the venues are voluntarily assigned by users of Foursquare at least a single category. And there are 7,753,274 occurrences of 833 unique categories identified in the dataset. Based on Foursquare's 3-level categorization system, we group the 833 categories into 8 coarse groups: Arts & Entertainment, College & University, Food, Great Outdoors, Home, Work and Other, Nightlife Spot, Shop, and Travel Spot. The distribution of the eight categories is listed in Table 1. Among the categorized venues, the category of Home, Work and Other presents almost one third (31.7%) of the venues. Venues in the category of Food (24.3%) and the category of Shop (17.1%) are also popular. The other five categories (Travel Spot, Great Outdoors, Nightlife Spot, Arts & Entertainment, and College & University) possess similar percentages (around 5% for each) in the dataset.

Besides the category information for venues, about 7.8% (i.e., 989,281) of the venues are labeled with at least a single tag. The tags are user-generated keywords posted by users of the location sharing service. Based on inspection, tags typically contain information such as category of the venue (e.g., coffee, food, and bar); items provided by the venue (e.g., burgers, flu shot, and long island iced tea); features of the venue (e.g., free wifi, 24 hrs, and pet friendly); location of the venue (e.g., houston downtown, bridge street); and users' comments for the venue (e.g., awesome, good deal, and great food). Each of the tagged venues is assigned with an average of 3.37 tags. Different from the categorization system, the tagging feature in Foursquare gives users more freedom to generate appropriate tags. In total, we find 615,457 unique tags that are collectively used a total of 3,329,641 times across all venues.

Together, these user-assigned tags and the top-level categories provide descriptive information about specific venues and provide clues to study the semantic correlation between venues. Recall that one of the key pre-processing steps in location-based search is category analysis of venues – to group together semantically-related venues – but in isolation we can see that the category assignments are fairly sparse (56%) and at a coarse-level; similarly, the tag information is even sparses (8% of all venues), and both tags and categories provide only *descriptive* information about the venues. Our

goal in the rest of the paper is to consider the traffic-driven temporal patterns revealed through checkins to augment this semantic grouping based on the real-world behaviors of users of these services.

4. EXPLORING SEMANTIC CORRELATION BETWEEN VENUES

In this and the following two sections, we begin an exploration of the temporal dynamics of venues as revealed through location sharing services. Given the large-scale checkin data, we propose to model venues through traffic-based patterns and seek to answer the following questions:

- Can we measure semantic correlation between venues based on associated traffic patterns?
- Can we cluster venues into semantically correlated groups based on traffic patterns?
- Can we use traffic patterns to accurately predict the semantic category for uncategorized locations?

We begin in this section by defining a traffic pattern and its frequency function. We discuss metrics to measure semantic similarity between traffic patterns, and we apply the temporal correlation measure to quantify the semantic relatedness between traffic patterns. Based on this initial study, we identify semantically correlated groups of venues based on measuring the pairwise temporal correlation between the venues' associated traffic patterns.

4.1 Modeling Venues

A **Traffic Pattern** (T Pattern) T for a venue over n time units is defined as the temporal dynamic of checkins during the time period. It can be measured by its **Frequency Function** F_T formally defined as $F_T = (f_{t_1}, f_{t_2}, ..., f_{t_n})$, in which f_{t_i} is the frequency for time unit t_i over the whole series of T. More specifically, for each venue, we generate a daily mean traffic pattern and a weekly mean traffic pattern given the timestamps of checkins in the venue. The **Daily (Mean) Traffic Pattern** contains 24 time units in which each of the time unit represents an hour in a day. Similarly, the **Weekly (Mean) Traffic Pattern** contains 70 time units in which each unit represents one tenth of a day. Examples of daily traffic pattern and weekly traffic pattern for Walmart are plotted in Figure 2 and Figure 3 respectively. The daily t pattern shows that customers tend to go shopping in Walmart in the afternoon and early evening, and the weekly t pattern indicates that there is a bigger crowd at Walmart over the weekends than on weekdays.

4.2 Temporal Similarity Measures

Given a traffic pattern for a venue, can we identify related venues based solely on this pattern? This is an important step for semantically grouping venues for improved location-based search. But perhaps traffic patterns do not vary much from venue to venue, meaning that traffic patterns could have only limited impact.

The most straightforward similarity measures for time-series data are Euclidean Distance [7] and its variants based on the common $L_p - norms$ (L_1 – Manhattan Distance, and L_2 – Euclidean Distance). These metrics can be easily implemented and are surprisingly competitive with other complex measures with a large training set. However, these distance measures are sensitive to noise and misalignments

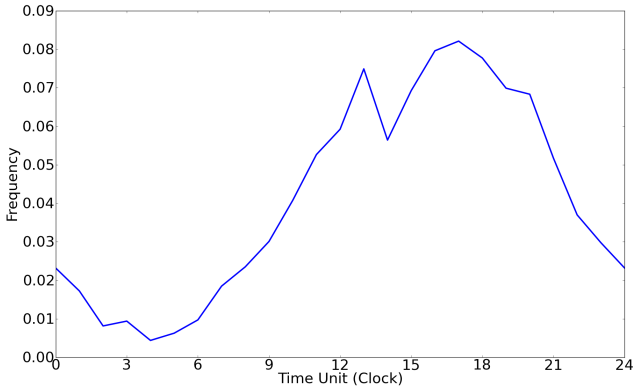


Figure 2: Daily Traffic Pattern for Walmart

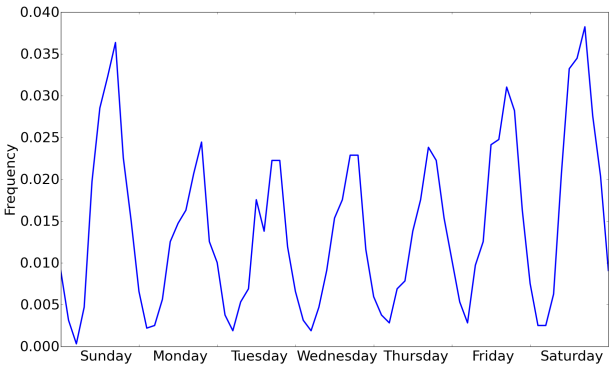


Figure 3: Weekly Traffic Pattern for Walmart

in time. Another effective temporal correlation measure is a temporally-grounded variation of the correlation coefficient. Given two traffic patterns Tp, Tq and their frequency functions F_{Tp} and F_{Tq}, the temporal correlation $T_{Corr}(F_{Tp}, F_{Tq})$ between the two traffic patterns Tp and Tq is:

$$T_{Corr}(F_{Tp}, F_{Tq}) = \frac{1}{n}\sum_i \left(\frac{f_{tp_i} - \mu(F_{Tp})}{\sigma(F_{Tp})}\right)\left(\frac{f_{tq_i} - \mu(F_{Tq})}{\sigma(F_{Tq})}\right)$$

where $\mu(F_{Tp})$, $\mu(F_{Tp})$ are the mean frequencies, and $\sigma(F_{Tp})$, $\sigma(F_{Tq})$ are the standard deviations for the two traffic patterns Tp and Tq. A version of this same metric was shown to be effective by Chien et al. [4] for measuring the similarity of search engine queries by comparing their frequency functions.

4.3 Mining Semantic Correlation between Venues

To study the semantic correlation between venues based on their traffic patterns, we sample a set of 271 venues from the checkins dataset with a criteria of at least 100 branches and 100 checkins to ensure the density of the traffic patterns. For each of the 271 venues, we retrieve both a mean daily traffic pattern and a mean weekly traffic pattern to capture their traffic. Then, we calculate the pairwise temporal similarity for all the pairs in the 271 venues set. After sorting the pairs of venues based on the descending order of temporal correlation measure, we find quite a few interesting pairs of venues that have obvious semantic correlation. Using the results for calculations based on daily mean traffic patterns only, we show the top-10 similar pairs of venues in Table 2. Each pair of venues in the table has obvious se-

Table 2: Top Pairs with Highest Temporal Correlation

Pair of Venues	Correlation
Target – Borders	0.949
Walgreens – CVS Pharmacy	0.947
Panda Express – Five Guys Burgers and Fries	0.947
Pizza Hut – California Pizza Kitchen	0.947
Chipotle – Five Guys Burgers and Fries	0.946
Staples – Apple Store	0.946
Target – Barnes & Noble	0.946
Subway – Jason's Deli	0.945
Chili's – Ruby Tuesday	0.944
Starbucks – Caribou Coffee	0.944

mantic correlation: both "Walgreens" and "CVS Pharmacy" are 24 hour pharmacy stores; both "Subway" and "Jason's Deli" are chain fast food restaurants; and both "Starbucks" and "Caribou Coffee" are coffee shops. The results listed in the table clearly indicate that the traffic pattern of a venue reveals its semantic category, and the temporal correlation between traffic patterns of two venues can help measure the semantic relatedness between venues.

Having all pairwise temporal similarities between venues, we are also interested to see whether we can find inherent groups of venues that belong to the same semantic category. For example, do all the coffee shops have similar temporal traffic patterns? We model the venues and temporal similarities as vertices and weights for edges in a graph. An edge between two vertices (venues) exists when the temporal correlation between traffic patterns of the two venues exceeds a pre-defined threshold. In this way, a graph modeling the semantic relationship between venues is generated. Instead of focusing on the whole graph itself, we are more interested in the strong connected components in the graph, which are potential candidates for semantic categories of venues.

As an example, when we set the pre-defined threshold for minimum temporal similarity as 0.93, a graph with 68 vertices, and 12 strong connected components is generated. Six example components are plotted in Figure 4. One component (plotted in Figure 4(a)) contains "Jason's Deli", "McAlister's Deli", "Qdoba Mexican Grill", "Subway", and "Zaxby's" which are all chain restaurants. The traffic patterns of the "steakhouse" component is displayed in Figure 4(b). Both the sub restaurants and the steakhouses have two peaks (lunch time and dinner time), though the frequencies differ dramatically. The major crowd arrives at sandwich shops at noon to grab lunch, while many more people choose to have steaks for dinner rather than lunch. Similar comparisons are plotted in Figure 4(c) and Figure 4(d), in which traffic patterns for the component of coffee shops and ice cream shops are featured respectively. People buy coffee in the early morning, and the busyness for the coffee shops gradually decreases during the day. However, people tend to have ice cream in the afternoon, and the ice cream shops are especially crowded in the early evening after dinner. Figure 4(e) and Figure 4(f) plot the components corresponding to office supply stores, and to a component of book store and pharmacies. Both the components have quite similar traffic patterns, and traffic patterns for book stores and pharmacies decrease from their peaks slower than the traffic patterns for the group of office supply stores. The other strong connected components are also highly semantically connected: there is a group of pizza restaurants, a group of juice shops, a group of chained family restaurants, and two groups of fast food restaurants. All the examples validate our hypothesis that measuring temporal correlation between traffic patterns can identify groups of venues that are semantically connected.

5. CLUSTERING VENUES

Motivated by the results shown in last section, a natural step to utilize the traffic patterns is to group the venues into similar categories. For example, clustering the traffic patterns may lead to clusters of categorized groups of venues, such as "coffee shops", "steakhouses", "hotels", and "gyms". Specifically, in this section, we apply several different clustering methods and different similarity metrics to cluster the venues based on features of traffic patterns.

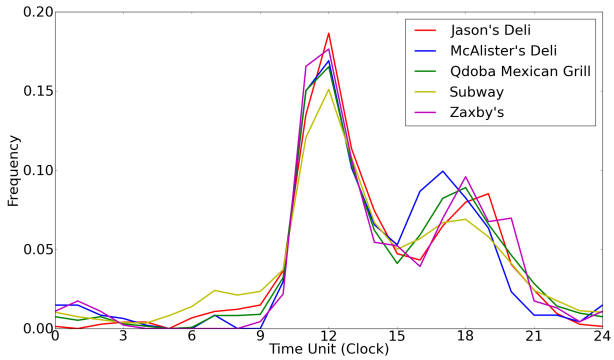
5.1 Methods

The graph modeling method we used in mining the semantically group of venues is one way to cluster the venues. However, it heavily relies on the value of the pre-defined correlation threshold, and it is difficult to control the number of strong connected components by tuning the threshold. Thus, in this section, we apply K-means and EM clustering algorithm for grouping the venues given their traffic patterns. We apply features of daily traffic pattern, weekly traffic pattern, and daily + weekly traffic pattern respectively in clustering the venues. We also explore using the tags for the venues to group the venues into semantically correlated categories.

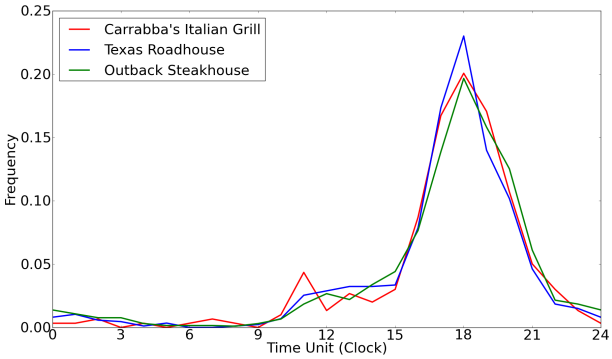
5.2 Experimental Setup

To evaluate the clustering results, a ground truth category is retrieved for each of the venues from the venues dataset collected from Foursquare. For K-means algorithm, we apply Euclidean Distance, Manhattan Distance, and Temporal Distance. Worth mentioning, the temporal correlation discussed earlier ranges between -1 and 1, with higher values indicating higher correlation. Since clustering requires a distance measure, we rescale the temporal correlation to define a metric **Temporal Distance**, where $T_{Dist} = 1.0 - \frac{T_{Corr}+1}{2}$. This temporal distance ranges between 0 and 1, with larger values indicating less similarity (greater distance). In evaluating the results for the clustering, we use the F1-Measure and Purity, which are both standard metrics to evaluate the quality of a clustering. Specifically, F1-Measure balances both the precision and recall of clustering. Purity measures the ratio of the total number of correctly clustered venues over the total number of venues.

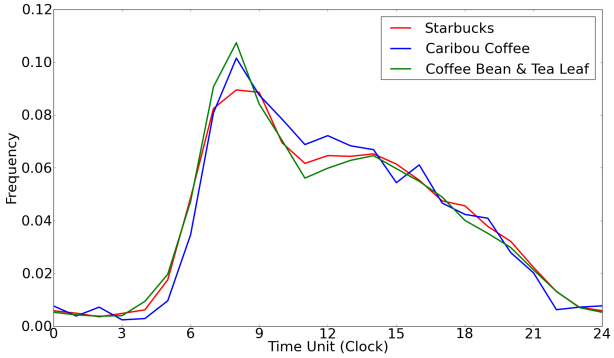
Additionally, four test sets are generated to evaluate the clusters based on the criteria of minimum number of checkins (500+, 300+, 200+, and 100+) for each venue. Venues with more checkins are expected to have denser traffic patterns, and thus contain stronger indication of semantic information. The four test sets include 148, 242, 383, 585 venues respectively. For each venue, two feature sets are generated: traffic patterns, and vector space models generated from tags. For the traffic patterns, we use daily traffic pattern, weekly traffic pattern, and daily plus weekly traffic pattern respectively. For the vector space models, we retrieve tags for the venues from the venue dataset, and generate features of tf, idf, and tf-idf values for the tags respectively. To normalize the features, we apply L2 normalization on both vectors of traffic patterns and for the vector space model. In the four testsets, only venues of the categories "Food", "Shop", "Home, Work, and Other", and "Travel Spot" (among the eight categories in categorization



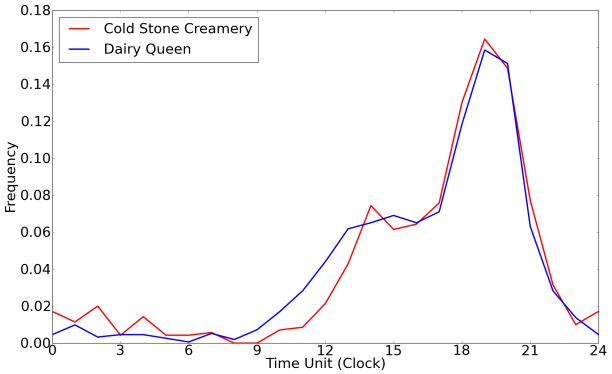
(a) Comparison between T Patterns of "Sub" Shops



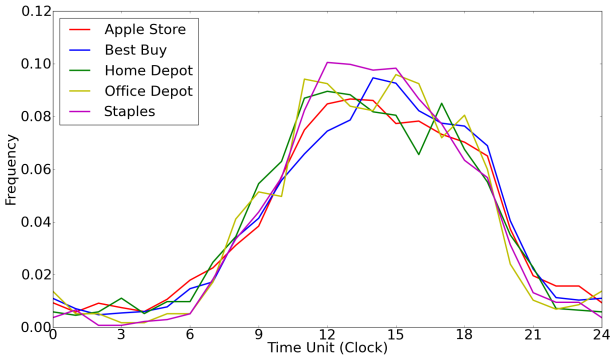
(b) Comparison between T Patterns of "Steakhouses"



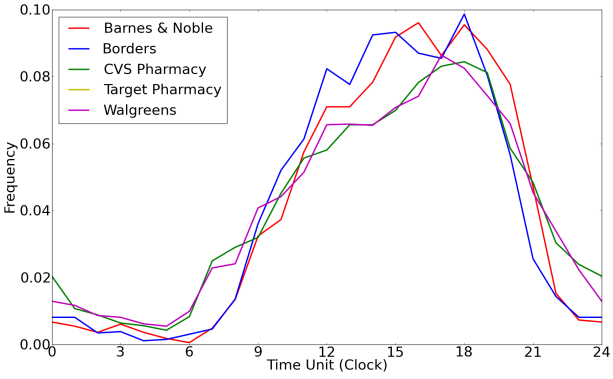
(c) Comparison between T Patterns of "Coffee" Shops



(d) Comparison between T Patterns of "Ice Cream" Shops



(e) Comparison between T Patterns of "Office Supply Stores"



(f) Comparison between T Patterns of "Book Stores and Pharmacies"

Figure 4: Comparisons between T Patterns in Different Strong Connected Groups

system of Foursquare) have sufficient checkins. Thus, in both K-means and EM algorithm, we pre-specify the number of clusters to be 4.

5.3 Experimental Results

Clustering with the Traffic Patterns:

Results for K-means and EM algorithms evaluated by F1-Measure and Purity using the four test sets are listed in Table 3. In most of the cases, clustering with the daily traffic pattern itself performs the best. Combining both daily and weekly traffic patterns performs better than using weekly traffic pattern alone. Among combination of method and metric, K-means plus Temporal Distance performs the best with a significant increase (around 15% to 20% increase) over K-means with the other two metrics, as well as the EM algorithm. K-means with Manhattan Distance performs better than K-means with Euclidean Distance. And they both outperform the EM algorithm when the venues have the sufficient checkins. However EM algorithms performs competitively when the venues have fewer checkins.

The best performing methods and their results are extracted from Table 3 and plotted in Figure 5. As we can see from Figure 5, generally, test sets with denser traffic patterns reach better performance in F1-Measure, and Purity. Results for the dataset with least checkins suffer from lack of sufficient data in traffic patterns. We also observe that the dataset with the most checkins do not reach the best performance. This is partly due to the lack of number of venues (only 2) in the category of travel spots.

Clustering with the Vector Space Models: As men-

Table 3: Results for Traffic Pattern Clustering

Dataset	Features	Daily T Pattern		Weekly T Pattern		Daily + Weekly T Pattern	
	Method / Metric	F1-Measure	Purity	F1-Measure	Purity	F1-Measure	Purity
500+ Checkins Dataset	K-means + Euclidean	0.495	0.595	0.419	0.534	0.412	0.520
	K-means + Manhattan	0.502	0.608	0.442	0.554	0.451	0.574
	K-means + T Distance	0.539	0.608	0.603	**0.689**	**0.608**	0.635
	EM	0.424	0.541	0.43	0.554	0.416	0.534
300+ Checkins Dataset	K-means + Euclidean	0.427	0.504	0.465	0.533	0.466	0.562
	K-means + Manhattan	0.502	0.566	0.435	0.533	0.476	0.583
	K-means + T Distance	**0.675**	**0.764**	0.586	0.674	0.616	0.698
	EM	0.465	0.537	0.467	0.579	0.47	0.562
200+ Checkins Dataset	K-means + Euclidean	0.416	0.483	0.463	0.441	0.456	0.535
	K-means + Manhattan	0.498	0.527	0.446	0.504	0.477	0.548
	K-means + T Distance	**0.671**	**0.700**	0.566	0.621	0.527	0.585
	EM	0.482	0.512	0.412	0.452	0.408	0.446
100+ Checkins Dataset	K-means + Euclidean	0.415	0.525	0.435	0.535	0.427	0.444
	K-means + Manhattan	0.437	0.535	0.406	0.504	0.437	0.515
	K-means + T Distance	0.571	0.667	0.552	0.658	**0.599**	**0.706**
	EM	0.477	0.568	0.403	0.487	0.464	0.405

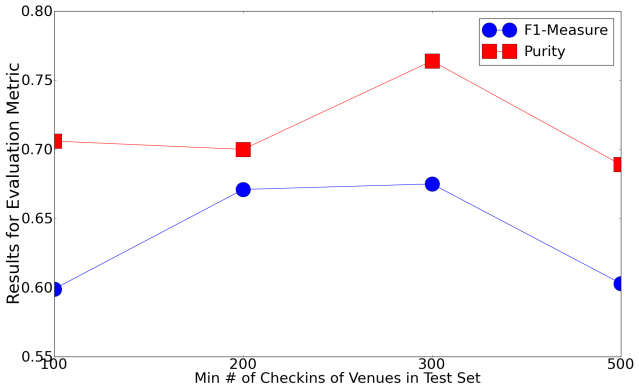


Figure 5: Comparison of Best Results over Different Test Sets

tioned, we retrieve tags for the venues in the four test sets from the venue dataset collected from Foursquare. We apply the same clustering methods using the features of vector space models (tf, idf, tf-idf) modeled from the tags respectively. However, for different methods over different training sets, all the venues tend to converge to a single large cluster, primarily due to the sparseness of the tags in the datasets.

Together, these results show that modeling venues by traffic patterns and using temporal correlation to measure venue similarity are viable alternatives to traditional content-based clustering methods.

6. SUPERVISED VENUE CATEGORIZATION

While clustering of venues by traffic-based temporal correlation can provide a foundation for organizing venues for improved location-based search, there is still the challenge of sparsity for venues on the "long tail". Hence, in this section, we propose and evaluate a traffic pattern-driven approach for supervised location categorization, wherein traffic patterns can be used to accurately predict the semantic category of uncategorized locations. Given a set of venues with known category labels, and their corresponding traffic patterns, are we able to train classifiers with the labeled set to categorize incoming venues with their corresponding traffic patterns? As in our study of clustering, we also consider an alternative tag-based model as a point-of-comparison.

6.1 Training Set and 10-Fold Cross Validation

The training set contains a set of the 271 most popular venues which all have at least 100 branches and over 100 checkins in the checkins dataset. As mentioned earlier, we retrieve the ground-truth labels for the venues from Foursquare venue data. The 271 venues are grouped into four categories (all belong to the category system mentioned before): Food; Home, Work & Other; Shop; and Travel Spot. We adopt the same two set of features as in the clustering work: traffic pattern (including daily traffic pattern weekly traffic pattern, and daily plus weekly traffic patterns), and vector space models (tf, idf, and tf-idf values) for tags associated to the venues.

We apply the features in training classifiers of Naive Bayes, 1NN (we iterate k from 1 to 5 for kNN, and 1NN always perform the best, so we fix k to 1 in the following experiments), AdaBoostM1, and SimpleCart. We also apply L2 normalization on both vectors of traffic patterns and for the vector space model. To evaluate the effectiveness of the classifiers, we use 10-fold cross validation and the F1-Measure.

Traffic Pattern Features:

Results for traffic pattern features are plotted in Figure 6. Using either daily traffic pattern or the weekly traffic pattern displays a strong indication of the category of the venue, reaching an F1-Measure higher than 0.8 with 1NN classifier. The other three classifiers – Naive Bayes, AdaBoostM1, and Simple Cart – seem incompatible with the time series data. Combining both daily traffic pattern and weekly traffic pattern gives a boost of 6.5% for Simple Cart, 1.7% for 1NN, and 0.6% for Naive Bayes. Overall, the 1NN classifier still performs best with an F1-Measure of 0.820 slightly above AdaBoostM1 and SimpleCart with a best F1-Measure of 0.819. These results agree with the observations in Xi et al.'s work [25] that 1NN classifier has excellent performance in time-series classification.

Vector Space Model Features:

Traditionally, an alternative way to categorize locations is using the social tags associated with the locations. As in the clustering approach, we again apply the vector space model features using tf, idf, and tf-idf respectively. Results in Figure 7 significantly differ from the results shown in Figure 6: for both tf and tf-idf, Naive Bayes, AdaBoostM1, and SimpleCart perform much better than 1NN, which verifies our previous assumption that those classifiers work well

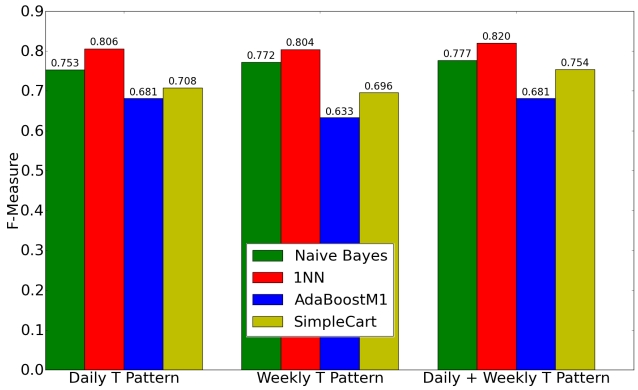


Figure 6: 10-Fold Cross Validation of 271 Venues Classification With T Patterns

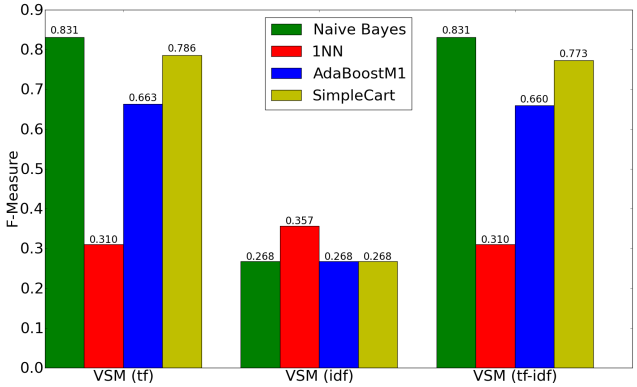


Figure 7: 10-Fold Cross Validation of 271 Venues Classification With Vector Space Models

with text-driven features. Classifiers trained by tf or tf-idf models significantly outperform ones trained by idf models. Furthermore, classifiers trained by tf-idf and tf have very similar results, which shows that enriching with the idf information could not help classify the venues. Besides, the best result for the vector space model based classifiers performs a little better (1.3%) than the best result for traffic pattern based classifiers.

Combination of the Two Sets of Features:

To answer the question of whether enriching information from tags could help facilitate venue categorization, we train classifiers on both sets of features (each set of features are normalized independently with L2-normalization before being merged). Unexpectedly, the results using both sets of features (daily + weekly t pattern, and tf-idf vector models) perform similar or even a little bit worse than the best performing classifiers trained by either set of features (results listed in Table 4). We attribute this drop in performance to the inclusion of possibly unhelpful countervailing features (94 temporal features, and 10,324 vector space model features). Thus, we apply feature selection to reduce the number of features by filtering irrelevant and redundant features.

Feature Selection:

We apply the standard Chi-Square feature selection method to reduce the number of vector space model features from 10,324 to 358 by filtering insignificant and redundant features. All traffic pattern features are considered significant

Table 4: 10-Fold Cross Validation of 271 Venues Classification With Traffic Pattern and Vector Space Models

Metric	Naive Bayes	1NN	AdaBoostM1	SimpleCart
F1-Measure	0.831	0.59	0.66	0.753

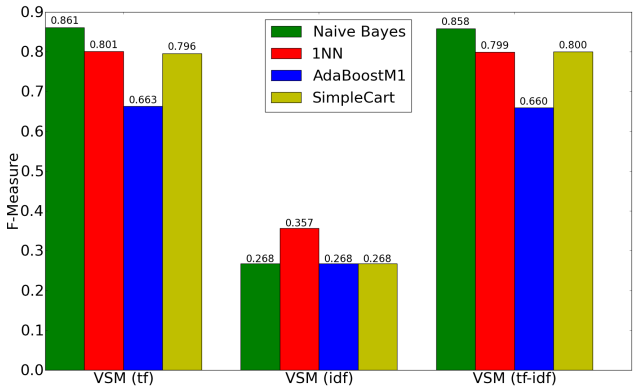


Figure 8: 10-Fold Cross Validation of 271 Venues Classification With Vector Space Models + Feature Selection

Table 5: 10-Fold Cross Validation of 271 Venues Classification With Traffic Pattern and Vector Space Model + Feature Selection

Metric	Naive Bayes	1NN	AdaBoostM1	SimpleCart
F1-Measure	0.866	0.765	0.66	0.75

by Chi-Square and so remain for the following classification experiments. Thus, we get exactly same results for traffic patterns comparing to results in Figure 6. The results for vector space model features are displayed in Figure 8. Comparing to the results without feature selection, classifiers trained by idf features, AdaboostM1 trained by tf features, and tf-idf features still perform the same. 1NN classifier trained with tf features and tf-idf features outperform with an almost 158.4% increase over the previous results. Besides, Naive Bayes classifiers trained with tf features and tf-idf features also have a 3.2% increase in their performance; as well as the Simple Cart with over 1% increase. The best performing classifier so far becomes Naive Bayes, which reaches a F1-Measure of 0.861.

Feature selection also shows its effectiveness when we train the classifiers using both the traffic pattern features and vector space model features. Comparing to results in Table 4, results in Table 5 show increase of performance for 1NN (29.7%), and Naive Bayes (4.2%).

6.2 Evaluation on Test Set

Based on the 10-fold cross validation on the training set, we only use 1NN as the classifier to classify venues based on the feature of traffic patterns. The test set of venues are generated based on the criteria of at least 10 branches with certain number of checkins above a pre-defined threshold. We set the threshold as 10, 30, 50, 100, 200, 300, and 500 respectively, and the corresponding number of venues in the test sets are listed in Table 6. As we can see from the table, with a relaxed criteria of only 10 checkins and above, the test set contains 1,392 venues, and with a strict criteria of

Table 6: Number of Venues in Test Sets

Min # of Checkins	10	30	50	100	200	300	500
# of Venues	1392	983	695	353	142	60	21

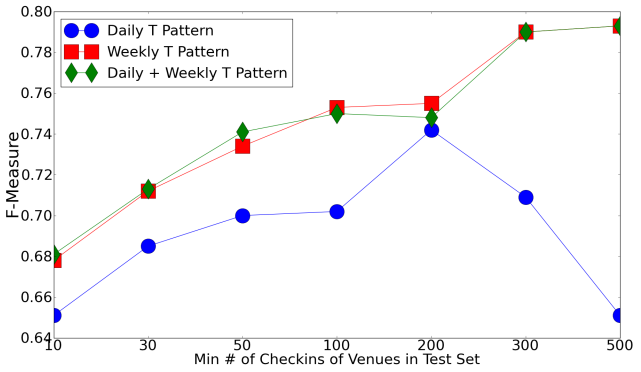


Figure 9: Evaluation of 1NN Trained by Traffic Patterns on Test Sets

500 checkins and above, the test set only contains 21 venues. Note that the test sets are disjoint with the training set.

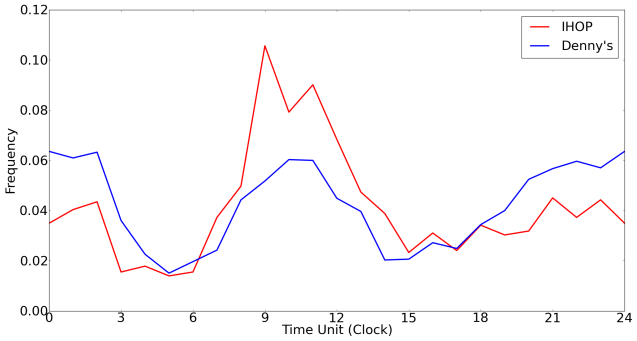
We train the 1NN classifier on the training set using daily traffic pattern, weekly traffic pattern, and daily plus weekly traffic patterns respectively, and evaluate test sets with corresponding features. The results for the classification are plotted in Figure 9 (each tick in x axis represents a test set). As we can see in the figure, with more checkins required for a venue in a test set, the results get better for the classifier trained by daily traffic pattern, and it reaches its peak with an F1-Measure of 0.742. However, it gets worse performance for the tests requiring at least 300 checkins and 500 checkins. This is probably caused by the lack of venues in the two test sets, and a small number of mis-classified venues can significantly affect the results. For the classifier trained by weekly traffic pattern and daily plus weekly traffic pattern, the results generally get better with test sets which only contain venues with dense traffic patterns. With weekly traffic pattern features, the classifier works much better overall than the one trained by daily traffic patterns. The classifier trained by daily plus weekly traffic patterns works slightly better than the one trained by weekly traffic patterns with test sets with relaxed condition, falls behind a little bit for test set 100, and test set 200, and finally catches up for test set 300 and test 500. In the figure, we can see that with only 50 or more checkins input per venue, the 1NN classifier can reach a F1-Measure almost 0.75, which shows its good performance in venue categorization.

7. AUGMENTING LOCATION-BASED SEARCH

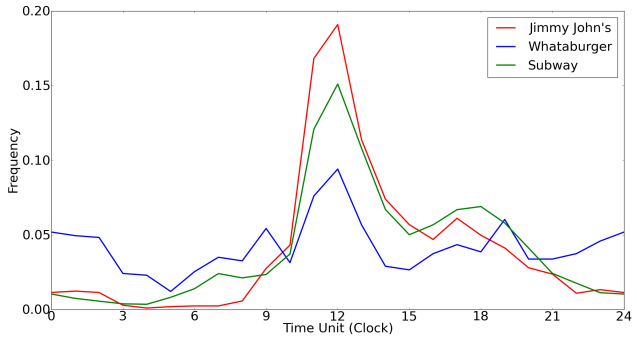
So far, we have seen that the traffic patterns for venues revealed through location sharing services contain rich information about the venues' semantic category. And we have successfully taken advantage of these traffic patterns for both unsupervised semantic group clustering and supervised venue categorization. In this section, we show how we can incorporate venues' traffic patterns and their category information into traditional location-based web search.

Answering Queries for Traffic:

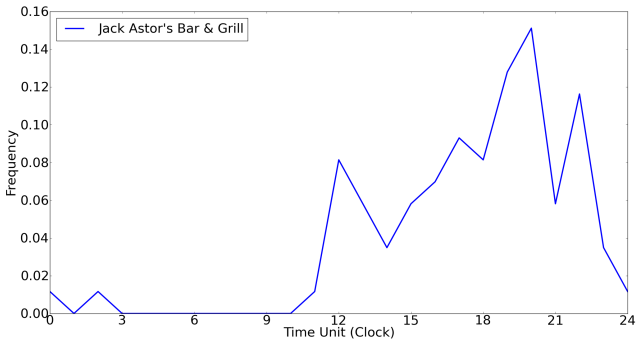
Traffic patterns and category information for venues can be easily incorporated into traditional location-based search to answer the information need for traffic. One scenario for answering the traffic-driven query is: Karen is searching for a restaurant which is off-peak during dinner time between 5 - 7 PM, so that she can enjoy the quiet environment talking with her friends. Knowing the traffic patterns and category for venues, the system could easily retrieve the venues nearby in the category of food, and rank the results by the descending order of busyness. Example results are plotted in Figure 10. For example, Karen can choose IHOP and Denny's where the crowd usually come in the early morning, lunch time, and late in the evening; she can also go to fast food venues like Jimmy Johns, and Chipotle which are crowded in during lunch time; besides, Karen can also choose grill & bars which are more popular in late evening like Jack Astor's Bar & Grill.



(a) T Patterns for IHOP and Denny's



(b) T Patterns for Jimmy John's, Whataburger, and Subway



(c) T Pattern for Jack Astor's Bar & Grill

Figure 10: Traffic Patterns for Venues Off-Peak between 5-7 PM

Location Recommendation based on Traffic:

Another potential application is recommendation of venues having similar traffic patterns. For example, Jerry plays a

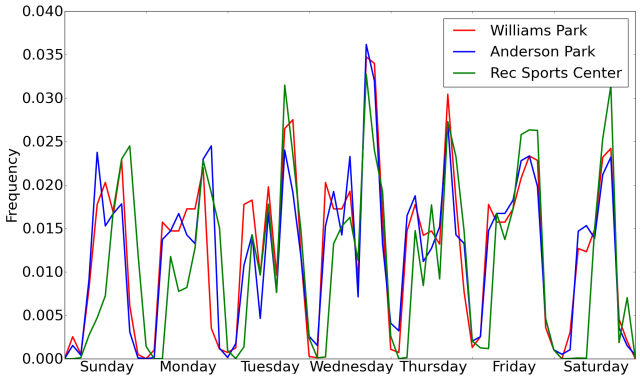


Figure 11: Example Showing Venue Recommendation based On T Pattern

lot of basketball, and tennis. He usually goes to the Williams Park during Wednesday early evening, and Saturday afternoon, which are both free time for him and peak times for guys to get-together and play basketball and tennis. Recently, he moves to a new neighborhood, and wants to find places nearby that have similar traffic patterns, so that he can meet new friends there and play some basketball or tennis. A traffic-driven location-based search can also easily handle this kind of queries. Given the name of the venue, the system calculates temporal similarity between traffic patterns of the venue and other venues in the same category (or in other categories as well), and return the locations with the highest temporal similarities. The example results are plotted in Figure 11, which shows the comparison of traffic patterns of Williams Park and two similar nearby venues Anderson Park and Rec Sports Center.

8. CONCLUSION

In this paper, we propose to mine traffic patterns revealed through location sharing services to augment traditional location-based search. Strong indication of semantic information are found in the traffic patterns generated from 22 million checkins from location sharing services. Then, we take advantage of the traffic patterns and successfully cluster venues into semantically correlated groups, and categorize incoming venues based on the associated traffic dynamics. Based on the results, we also provide two examples to show how traffic-driven semantic organization of locations may be naturally incorporated into traditional location-based search.

9. REFERENCES

[1] S. Asadi, C. Chang, X. Zhou, and J. Diederich. Searching the world wide web for local services and facilities: A review on the patterns of location-based queries. In *Advances in Web-Age Information Management: 6th International Conference*, WAIM '05, 2005.

[2] O. Buyukkokten, J. Cho, H. Garcia-Molina, L. Gravano, and N. Shivakumar. Exploiting geographical location information of web pages. In *Proceedings of the ACM SIGMOD Workshop on the Web and Databases*, 1999.

[3] Z. Cheng, J. Caverlee, and K. Lee. Exploring millions of footprints in location sharing services. In *ICWSM 2011*.

[4] S. Chien and N. Immorlica. Semantic similarity between search engine queries using temporal correlation. In *WWW 2005*.

[5] M. Ciaramita, V. Murdock, and V. Plachouras. Online learning from click data for sponsored search. In *WWW 2008*.

[6] J. Ding, L. Gravano, and N. Shivakumar. Computing Geographical Scopes of Web Resources. In *VLDB 2000*.

[7] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast subsequence matching in time-series databases. In *SIGMOD 1994*.

[8] T.-c. Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24, 2011.

[9] Q. Gan, J. Attenberg, A. Markowetz, and T. Suel. Analysis of geographic queries in a search engine log. In *Proceedings of the first international workshop on Location and the web*, LOCWEB '08, 2008.

[10] S. A. Golder, D. M. Wilkinson, and B. A. Huberman. Rhythms of social interaction: Messaging within a massive online network. In *Proceedings of the Third Communities and Technologies Conference*, 2007.

[11] T. Joachims. Optimizing search engines using clickthrough data. In *SIGKDD 2002*.

[12] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46, 1999.

[13] A. Kulkarni, J. Teevan, K. Svore, and S. Dumais. Understanding temporal query dynamics. In *WSDM 2011*.

[14] W. T. Liao. Clustering of time series dataâĂŤa survey. *Pattern Recognition*, 38(11), 2005.

[15] J. Lindqvist, J. Cranshaw, J. Wiese, J. Hong, and J. Zimmerman. I'm the mayor of my house: Examining why people use foursquare - a social-driven location sharing application. In *SIGCHI 2011*.

[16] A. Markowetz, Y.-Y. Chen, T. Suel, X. Long, and B. Seeger. Design and implementation of a geographic search engine. In *Proceedings of the ACM SIGMOD Workshop on the Web and Databases*, 2005.

[17] K. S. McCurley. Geospatial mapping and navigation of the web. In *WWW 2001*.

[18] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil. An empirical study of geographic user activity patterns in foursquare. In *ICWSM 2011*.

[19] K. Radinsky, E. Agichtein, E. Gabrilovich, and S. Markovitch. A word at a time: Computing word relatedness using temporal semantic analysis. In *WWW 2011*.

[20] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. In *Information Processing and Management*, volume 24, 1988.

[21] M. Sanderson and J. Kohler. Analyzing geographic queries. In *SIGIR 2004*.

[22] O. Uryupina. Semi-supervised learning of geographical gazetteers from the internet. In *Proceedings of the HLT-NAACL Workshop on the Analysis of Geographic References*, 2003.

[23] C. Wang, X. Xie, L. Wang, Y. Lu, and W.-Y. Ma. Detecting geographic locations from web resources. In *GIR '05: Proceedings of the 2005 workshop on Geographic information retrieval*, 2005.

[24] C. Watters and G. Amoudi. GeoSearcher: location-based ranking of search engine results. *Journal of the American Society for Information Science and Technology (JASIST)*, 54(2):140–151, 2003.

[25] X. Xi, E. Keogh, C. Shelton, and L. Wei. Fast time series classification using numerosity reduction. In *ICML 2006*.

[26] G.-R. Xue, H.-J. Zeng, Z. Chen, Y. Yu, W.-Y. Ma, W. Xi, and W. Fan. Optimizing web search using web click-through data. In *CIKM 2004*.

[27] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *WSDM 2011*.

[28] M. Ye, P. Yin, and W. C. Lee. Location recommendation for location-based social networks. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2010.