SemanticTraj: A New Approach to Interacting with Massive Taxi Trajectories

Shamal AL-Dohuki
Kent State University
Farah Kamw
Kent State University
Ye Zhao
Kent State University
Chao Ma
Kent State University
Yingyu Wu
Kent State University
Jing Yang
UNC Charlotte
Xinyue Ye
Kent State University
Fei Wang
Zhejiang University
Xin Li
China Petroleum University
Wei Chen
Zhejiang University

Abstract—Massive taxi trajectory data is exploited for knowledge discovery in transportation and urban planning. Existing tools typically require users to select and brush geospatial regions on a map when retrieving and exploring taxi trajectories and passenger trips. To answer seemingly simple questions such as "What were the taxi trips starting from Main Street and ending at Wall Street in the morning?" or "Where are the taxis arriving at the Art Museum at noon typically coming from?", tedious and time consuming interactions are usually needed since the numeric GPS points of trajectories are not directly linked to the keywords such as "Main Street", "Wall Street", and "Art Museum". In this paper, we present SemanticTraj, a new method for managing and visualizing taxi trajectory data in an intuitive, semantic rich, and efficient means. With SemanticTraj, domain and public users can find answers to the aforementioned questions easily through direct queries based on the terms. They can also interactively explore the retrieved data in visualizations enhanced by semantic information of the trajectories and trips. In particular, taxi trajectories are converted into taxi documents through a textualization transformation process. This process maps GPS points into a series of street/POI names and pick-up/drop-off locations. It also converts vehicle speeds into user-defined descriptive terms. Then, a corpus of taxi documents is formed and indexed to enable flexible semantic queries over a text search engine. Semantic labels and meta-summaries of the results are integrated with a set of visualizations in a SemanticTraj prototype, which helps users study taxi trajectories quickly and easily. A set of usage scenarios are presented to show the usability of the system. We also collected feedback from domain experts and conducted a preliminary user study to evaluate the visual system.

Index Terms—Taxi Trajectories, Taxi Document, Textualization, Name Query, Semantic Interaction, Text Search Engine

1 Introduction

Advanced sensing technologies and computing infrastructures have produced a variety of trajectory data of humans and vehicles in urban spaces. Taxi trajectory data records realtime moving paths sampled as a series of positions associated with vehicle attributes over urban road networks. Massive trajectory data contains abundant knowledge about a city and its citizens which has been widely used in urban computing [40]. Exploratory visualization systems are demanded to study taxi trajectories with efficient user interaction and instant visual feedback. However, users often need to select, brush, and filter regions on maps to interact with GPS points and trajectory paths. Complex operations are needed to complete some straightforward tasks. We give two scenarios as examples:

- Scenario 1: A shopping mall has a plan to open shuttle buses for their customers. Its manager wants to investigate where and when visitors take taxis to the mall.
  Task 1: "For the taxi trips arriving at the shopping mall, what are their major pick-up locations? "
- Scenario 2: Two witnesses reported a criminal suspect taking a taxi passing South Street and North Street between 3pm and 3:20pm. A policeman wants to find suspicious taxi paths.
  Task 2: "What are the taxi trips passing South Street and North Street between 3pm and 3:20pm? What are the other streets/POIs they visited?"

For Task 1, the manager needs to: (1) select a region enclosing the mall on the map to display drop-off points; (2) select a given time period; (3) brush to filter those points related to this mall inside this region; (4) display the corresponding pick-up points of passenger trips on the map; (5) observe the map to find hot locations with a high density of pick-up points; (6) find the streets and POIs of these hot locations. Then, the candidate streets/POIs for shuttle stops are found.

For Task 2, the policeman needs to (1) select a region around North Street on the map; (2) select the time period from 3pm to 3:20pm in a time selection tool (e.g., a slider); (3) brush to select all GPS sample points residing on North street; (4) display the GPS points in the same taxi trips with the selected points from (3); (5) repeat steps (1)-(3) to select points on South Street. After these steps, the required taxi trips are found; (6) Find other streets they also passed.

Apparently, these interaction steps require non-professional users to be trained for the map-based operations. Advanced trajectory filtering tools [2] are designed where users can visually operate specific filters such as lens on maps (e.g., [20, 24]). The learning curve may deter some domain or public users. Can we help the users complete the query tasks in an alternative way, akin to querying over keywords in a search engine? For example, can the policeman simply input "Select trips on North Street AND South Street AND [3pm, 3:20pm]" for Task 2? Then, the resulting trips are visualized on maps together with intuitive text labels, or through a short textual summary, so that the policeman can immediately find the requested street names. Such interactions are natural and easy to conduct for city residents and general practitioners.

- *Shamal Al-Dohuki, Farah Kamw, Ye Zhao (corresponding author), Chao Ma, and Yingyu Wu are with the Department of Computer Science, Kent State University. Xinyue Ye is with the Department of Geography. Email: zhao@cs.kent.edu; fkamw,sadohuki,cma1,ywu23,xye5@kent.edu*
- *Jing Yang is with the Department of Computer Science, University of North Carolina at Charlotte. E-mail: jyang13@uncc.edu*
- *Fei Wang and Wei Chen are with the Department of Computer Science, Zhejiang University. E-mail: wolflyecn@gmail.com, chenwei@cad.zju.edu.cn; Xin Li is with the China Petroleum University.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

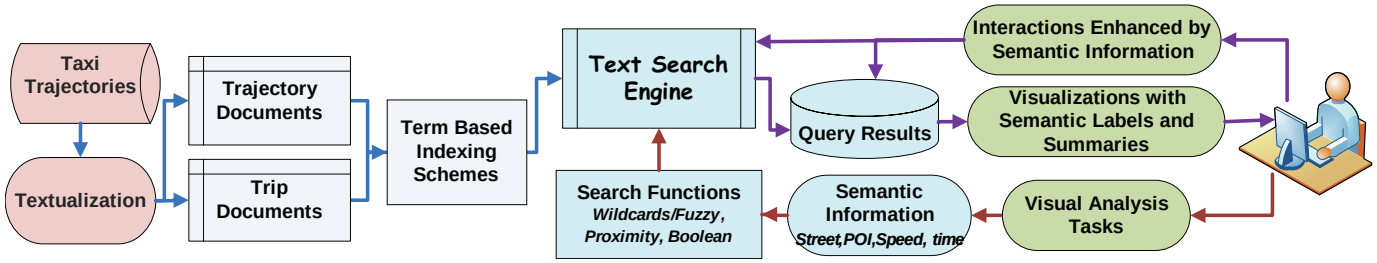


Fig. 1. *SemanticTraj framework of processing, searching and visualizing taxi trajectory data.*

In this paper, we develop SemanticTraj to enable such new interactions for studying taxi trajectory data. It supports non-professional users to retrieve taxi trips by giving street or POI (Point of Interest) names, speed descriptions, and their boolean combinations. These terms are referred to as *semantic information* in this paper. Then, users can visually study the results with a set of visual representations enhanced by the semantic information.

Most people are familiar with the process of phrasing queries as keywords to dig up information. In SemanticTraj, the experience is extended to taxi trajectory data, leading to a new paradigm of the interaction between users and trajectory data, in addition to selection or brushing over maps. In particular, taxi trajectory data is transformed into "taxi documents" by a process of "textualization", which projects a GPS point to a street or POI name, and maps a numerical speed value into a descriptive term (e.g., slow, normal, fast). The semantic information gives direct cues of geographical and transportation information for users living in the city. Users are allowed to retrieve matching data from keywords of interest using text search engine (Apache Lucene in our experiments). Moreover, the data management and query performance is highly efficient in the engine with well-designed index schemes. Therefore no specific data structures are needed to manage trajectory data over grids and trees. Interactive visualization is well supported by the fast computation over big taxi data. Furthermore, users can flexibly query taxi documents by wildcard, boolean, fuzzy, proximity and range search with semantic hints. To the best of our knowledge, our approach is the first to employ text search engine in managing and querying taxi trajectories.

The semantic information of query results is integrated with visual representations and interactions to promote easy understanding. First, textual labels are added on taxi traces and streets in the map view. It is well known that text/title information is the most effective visual cues for users [6]. Users reading the labels can immediately get ideas about the search results. Second, a meta-summary of the taxi trajectories is provided to present their mobility features. Each individual trajectory also has a meta-summary of its own behavior. Users can directly refine the query results by interacting with keywords in the meta-summaries.

In summary, our major contribution is a novel approach for effective interaction with big taxi data. First, our approach uses textualization to map trajectory data to taxi documents. Second, our approach manages and queries the taxi documents in a text search engine. Third, we have developed a fully working prototype for this approach. It allows users to visually explore trajectories with enriched semantic information. A set of usage scenarios are presented to illustrate its usability.

2 Related Work

Our project is related to the mining and visual analysis of trajectory data. It also provides a new way of spatial data management. Meanwhile, semantic information is utilized in our system. Therefore, we introduce the related work in these three directions.

2.1 Trajectory Data Mining and Visual Analysis

Trajectories of human and vehicle motions are one of the most important data used in urban data analytics [42]. Utilizing the trajectory data has been categorized into three main categories: the study of the collective behavior of a city's population, the traffic flow, and the operators (e.g. drivers) [8]. For instance, vehicle trajectory data has been used in traffic monitoring and prediction [26], personalized route recommendation [13], urban planning [41], driving routing [34, 35], extracting geographical borders [29], service improvement [36], and energy consumption analysis [37]. Public transit trajectories are used in bus arrival time predictions [44], user's transportation mode inference [44], and travellers' spending optimization [21].

The analytical tasks are supported by visualizing the spatial-temporal data [2], where the techniques combine map-based displays and information visualization techniques [23]. Andrienko et al. [1] transformed GPS-tracked trajectories into aggregated flows between areas to depict important moving patterns over a city. Wang et al. [33] explored traffic data recorded on sparsely distributed cells in a city. Liu et al. [22] presented a visual analytics system of route diversity. Wang et al. [32] presented an interactive analysis system for urban traffic congestion. Pu et al. [28] visually monitored and analyzed complex traffic situations in big cities. Wang et al. [31] presented a visual reasoning approach for the data-driven transport assessment on urban roads. It supports dynamic query of traffic situations within a bi-directional hash structures on a spatial grid. Ferreira et al. [15] allowed users to visually query taxi trips on spatio-temporal constraints, which was designed only for queries over the pick-up and drop-off pairs.

2.2 Spatial Trajectory Management

Spatial indexing mechanisms such as minimum bounding rectangles (MBR) and hierarchical trees [25] are often used over a spatial division. Indexing trajectory data requires extra work in maintaining the links between sampling points. R-tree [17] and its extensions (e.g., [14]) build MBRs based on clusters of trajectory segments. Other methods maintained the linkage of samples and time indexes over fixed grid cells (e.g. over quadtree) (e.g. [9, 12]). Sample points can be hashed [31, 43] on a cell to achieve a fast query. Our method is distinguished from them by managing trajectories as textual documents and performing queries over the documents where no spatial MBRs or trees are needed.

Managing massive trajectories with traditional methods requires careful design and optimization of computer systems. For instance, SETI [9] and TrajStore [12] store trajectory segments that are spatially close on the same disk partitions and memory pages to accelerate query processing. Our method instead utilizes the data management capability of the text search engines, where the system level data optimization is well-developed including memory and disk use, caching, encoding and compression [7].

2.3 Semantic Trajectory Management

Many approaches enrich trajectories by associating locations with semantic meanings such as labelling user activities or place names (e.g., Restaurant, Theater). A survey paper well discussed current approaches for modeling and analysis of semantic trajectories [27]. The concepts of semantic subtrajectories, points, geographical places, events, etc. were proposed in a concept model [5]. A knowledge-based framework was also presented for semantic enrichment and analysis of moving data [16]. Activity trajectory similarity query (ATSQ) [39] supported searching activities through a specific grid-index combined with inverted index of a predefined set of activities. A hybrid index structure (GIKI) further organized the trajectories with both spatial and textual similarity [38]. These methods partitioned trajectories into segments by considering spatial, temporal and semantic features. They

were designed for finding trajectories with predefined activity terms in regions. Our aim is different which is to visualize trajectories by exact street names, status, and time ranges. Streets become the major query units instead of regions and the search is enabled by text search engine after mapping GPS points to street names.

Su et al. [30] generated a short descriptive text of a trajectory by selecting features of trajectory segments. Annotated semantics was also used for automatic insight externalization [10]. Chu et al. [11] studied massive trajectories as a document corpus based on similar transformation where LDA topic modeling is employed. However, their work found hidden themes of population movement and did not provide query functions to retrieve data.

3 Overview

3.1 Taxi Trajectory Data

Massive trajectory datasets are acquired by taxis traveling over cities. A taxi trajectory records consecutive samples at an interval of a few seconds in a given time period. Each sample includes the GPS location (longitude and latitude), vehicle ID, speed, time, occupancy status, direction, and possibly other attributes. A taxi trip consists of those samples when the taxi was hired by customers. It then refers to one taxi service trip for passengers.

3.2 Visual Exploration Tasks and Our Approach

SemanticTraj allows users to retrieve and visually explore a taxi trajectory dataset such as:

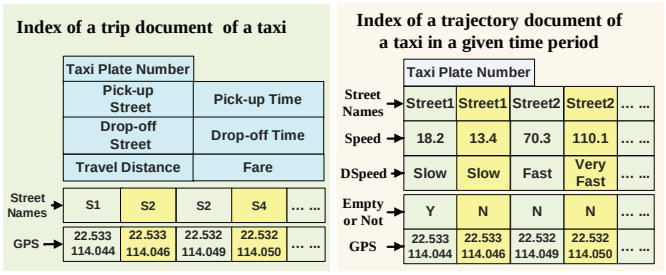
- *Task1: Taxi trips passing a street in a given time period;*
- *Task2: Taxi trips passing multiple streets in given time periods with different logic conditions;*
- *Task3: Taxi trips with single or multiple POIs;*
- *Task4: Taxi trajectories passing given streets and POIs;*
- *Task5: Taxi trajectories with specific behaviors in travel speed (e.g., slow, fast, change from slow to fast).*

Figure 1 shows the framework of SemanticTraj. First, raw taxi trajectories are processed and transformed into taxi documents through "textualization", which projects geographic locations to streets, and maps speed values to descriptions. A taxi document can be a trip document consisting of the streets a taxi passed with passengers. It can also be a trajectory document including all streets a taxi traversed in a time period. The trajectory dataset of all taxis is converted to a taxi document corpus. Indexing schemes are designed for the corpus to manage the taxi documents using a text search engine. Then with an interactive visual system, users query the data by a variety of convenient text search functions, and the query results are visually presented. Rich semantic information is conveyed through semantic labels and meta-summaries, which enable easy and fast understanding of the query results. Users can operate on the visual interface to refine and explore the results for insight discovery.

4 Textualization of Taxi Trajectories

We refer the process of converting an attribute in the raw data to a text term as "textualization". In general, the aim is to create semantic-rich forms by externalizing associated contextual information of the original data. Domain knowledge and user input are then incorporated into the transformed data. We transform massive taxi data in the following ways:

1. Each geographical location of latitude and longitude is mapped to the street name it resides. Such a transformation process is implemented through road matching to find the closest street of a given position where we follow the implementation in [11]. As a result, a sequence of GPS points over a trajectory are mapped into a taxi's traversed streets.



(a) (b)

Fig. 2. *Indexing of taxi documents. (a) An index of a trip document. (b) An index of a trajectory document.*

2. The associated attributes are transformed to semantical information. The numeric travel speed is converted to a descriptive term as:

   - speed$<$0.01Km/h $\mapsto$ *Stop*,
   - 0.01Km/h$\leq$speed$<$20Km/h $\mapsto$ *Slow*,
   - 20Km/h$\leq$speed$<$60Km/h $\mapsto$ *Normal*,
   - 60Km/h$\leq$speed$<$100Km/h $\mapsto$ *Fast*,
   - 100Km/h$\leq$speed $\mapsto$ *Very Fast*.

   In this way, text search can quickly find specific behavior described by the terms, such as identifying speeding drivers with *Very Fast*. Such a mapping is currently predefined after consulting local drivers. It can be configured by domain users based on their specific knowledge and aims in practical applications.

5 Taxi Documents

By introducing semantic meanings into the trajectory data, we create *taxi documents* from massive trajectories on which text based searches are applied. Taxi documents enable us to manage and search big trajectory data using text search engines. In an engine, each document of a corpus is represented by an *index* consisting of multiple fields, where each field stores a term. A term usually refers to words but may also be a date, number, etc, whereas they are represented in a textual form. The query criteria combine a set of given terms in the corresponding fields. Next we discuss the trajectory and trip documents in Sec. 5.1 and 5.2, respectively. Then we summarize the search functions in Sec. 5.3.

5.1 Trip Documents

Fig. 2a shows an index of one *trip document* related to one trip of a taxi's service to passengers. The index includes the fields of Pick-up and Drop-off streets and Times, and the attributes associated with this taxi trip such as its Travel distance and Fare. The field of Street Names stores the sequence of streets the taxi traveled. In particular, the field of *GPS* is used to store the sequence of GPS points, which is used to draw the trajectory in visualization. If the whole trip path is not considered, it might only store the pick-up and drop-off GPS locations.

A set of such indexes are generated for a collection of trip documents in a time period (e.g., one day). These indexes are stored in one index file, C. Multiple index files can be queried simultaneously to search trips in multiple periods (days). The analysis tasks in Sec. 3.2 are conducted by text search engine using specific query conditions. Two examples are:

- *Task1 Question*: What are the pick-up and drop-off locations for the trips passing S?
  Query Condition: $\{S$ *in Street Names in* $C\}$.
- *Task2 Question*: What are the trips picking up passengers at S_1 during T_1 AND dropping them off at S_2 during T_2?
  Query Condition: $\{\{S_1$ *in Pick-up Street AND Pick-up Time in* $T_1\}$ *AND* $\{S_2$ *in Drop-off Street AND Drop-off Time in* $T_2\}\}$.

If the tasks are about a POI P (*Task3*), we first find the pre-computed street segments close to P, and then apply similar street-based queries. More complex queries with combined constrains and logic operators can be implemented in a similar manner. For example, we can add $\{$*Fare:[30 TO 50]*$\}$ in the above queries to get trips earning between 30 and 50.

5.2 Trajectory Documents

Fig. 2b shows an index of one *trajectory document* related to one taxi's trajectory in a given time period T. The field of Street Names includes the sequence of streets the taxi traveled during T. The field of DSpeed stores the description terms (e.g., slow, fast) of textualized numeric speeds. The Occupation Status is represented by Y or N at each point.

Using such an index for each trajectory, a document corpus is created and stored in one index file $C(T)$. Using the length of T as 10 minutes, a set of 144 indexing files jointly represent the data in a whole day. Two example tasks they support are:

- *Task4 Question*: Given all the taxi trajectories passing S during T, what is their average speed? How many are occupied?
  Query Condition: $\{$*S in Street Names in* $C(T)\}$.
- *Task5 Question*: What are the taxis whose speed shows an abrupt speed-up from slow to very fast (across normal) during T?
  Query Condition: $\{$*Slow VeryFast* $\sim$*1 in DSpeed in* $C(T)\}$.

Here a proximity search *Slow VeryFast* $\sim$*1* is utilized to find taxis whose speed suddenly changes from slow to very fast (see the search functions and syntax in Sec. 5.3). In a similar manner, we can search a long phrase *Slow Slow ... Slow* whose appearance may reflect potential traffic congestion, or search *VeryFast VeryFast ... VeryFast* for finding taxis who may have excessive speeding. Another interesting example is to search consecutive street names *Road1West Road2North*, which reflects a left turn. The returned trajectories may violate the traffic law if such a left turn is not allowed. It is the textualization and text search engine which provide a novel tool for users to conduct these jobs.

The indexing scheme of trajectory documents is different from that of trip documents, because of their different analytical focuses. Users study trajectories to learn traffic situations, while they are mostly interested in taxis' behaviors when investigating trips. Due to such differences, we use the time-period based indexing approach $C(T)$ for trajectory documents. Users can query the trajectories in multiples of 10 minutes. Surely T can be further reduced for more refined data search which increases the number of index files. The benefit is that this indexing scheme supports quick data feedback and visualization. Instead, if one big index file for 24 hours is used, we will need to read through all the returned documents and filter the results for a given small time period, where the extra computation reduces time efficiency. In summary, the design of indexing schemes of taxi documents can vary according to the analytics tasks to be performed.

A text search engine maintains and manages an inverted index structure based on the indexing scheme of documents. It is designed to efficiently construct, compress, and manipulate all the document indexes, as well as the inverted index structures (see [7] for details). For example, the Apache Lucene engine is deliberately tuned for scalable, high-performance indexing [4]. The index size is roughly 20-30% of the size of documents indexed. It is also optimized over memory and disk to handle data indexing and queries. Utilizing the engine gives us the power and flexibility of data management and interactive retrieval for efficient visualization.

5.3 Flexible Search Functions

Taxi documents enable a new means of interaction between users and taxi trajectories. Users can utilize flexible search functions combining street/POI names and speed descriptions. Boolean, wildcard, fuzzy, proximity and range queries are supported over taxi documents.

- *Boolean Query:* A boolean query combines multiple queries of individual terms with boolean conditions. It allows users to quickly conduct a task (e.g., *Task2*) which retrieves trajectories by multiple conditions.

Table 1. Syntax of Queries.

Type	Syntax	Description
Term Query	t	Find term t
Phrase Query	"$t_1 ... t_n$"	Find ordered terms in a given phrase
Wildcard Query	s_1*s_2	Find terms leading by string s_1 and ending with s_2
Fuzzy Query	$t\sim$	Find term t approximately
Proximity Query	"$t_1 ... t_n$"$\sim d$	Find phrase within distance d
Field Query	f:Q	Find a query Q in field f
Range Query	f:[t_1 TO t_2]	Find terms lexicographically between t_1 and t_2
Boolean Query	Q_1 OR(AND,NOT) Q_2	Find Q_1 or (and, not in) Q_2

Table 2. Query performance on trip documents.

Time Period	Index Size	Indexing Time (sec)	Q1 Time (sec)	Q1 Hits	Q2 Time (sec)	Q2 Hits	Q3 Time (sec)	Q3 Hits
One Day	297MB	9	0.15	25k	0.14	6k	0.19	39k
One Week	1.00GB	336	0.21	90k	0.19	19k	0.32	137k
One Month	6.55GB	2,012	1.75	590k	1.32	141k	2.85	918k

- *Range Query:* A range query matches the documents whose terms fall into the supplied range. For example, we can query taxi pick-up time between *[07:00:00 TO 10:00:00]*, or query fare of taxi trips larger than 30.
- *Wildcard and Fuzzy Query:* A wildcard query supports users with single and multiple missing characters in query terms. Users can query a street without clearly remembering the name in full, such as *north** for *north ave* or *north str*. A fuzzy query finds any matched terms with Levenshtein Distance. For example, *north*$\sim$ can find terms like lorth or nortn.
- *Proximity Query:* A proximity query supports matching words within a specific distance in text. For example, a query of "*slow veryFast* $\sim$*1*" finds those documents where *slow* and *fast* happen within two consecutive words. The query helps users immediately find a speed change event from *slow* to *fast* beyond *normal*.

These functions flexibly support visual analysis tasks, which may require complex operations in traditional region queries with geo-spatial indexing. Table 1 summarizes the query syntax. The complete syntax and examples can be found in [3]. Users with knowledge of Boolean operations need a small effort to become familiar with the query language. For novice users, we further design a form to fill in the query fields without phrasing the queries.

The indexing model also facilitates ranking query results by customized scoring. Taxi trips can be ranked by the term frequency of a street name (S), which implies how many samples are on S in a taxi trip. This score reflects a trip spending lots of time on S. The ranking can also be completed by the length or the fare of a taxi trip, so that users immediately get the longest trip or the trip achieving the highest fare. The ranking scheme is very useful in visualization tasks to provide users preferred information of query results.

6 Data Processing and Query Performance

We use the taxi trajectory data of Hangzhou, China. Hangzhou has a population of about 2.5 million and taxi is one of its major transportation methods. The dataset of a whole month (Dec. 1-31, 2011) has a raw size of 77GB acquired by 8,120 taxis. Each day there is a raw data size around 2.5GB in the format of raw GPS sample points and associated attributes (ID, speed, time, etc.). Data preprocessing consists of several steps: (1) Remove erroneous (duplicated) records. (2) Add street names and speed descriptions to the points after performing textualization (which needs map matching). (3) Create trajectories from the points with the same taxi ID. Here the points should be sorted by time since in a trajectory they need to be stored in sequence. For fast computing, we sort and generate trajectory segments for each small time interval (10 mins) of a day, and then join these segments for complete daily trajectories. (4) Find trips from the trajectories according to the occupancy status. The total processing time is around 9 hours for the dataset. This speed can be accelerated by using an advanced map-matching method and parallel processing of the points and trajectories. In results, the size of the trajectory documents is about 38GB,

in which 10% is used for names. The size of the trip documents is around 8.5GB with 6,734,497 trips. Next we show the performance of name queries using Lucene with this dataset tested on a 64bit Windows 7 workstation (Intel Xeon E5620 2.40GHz, 24GB, 1TB).

Three queries are applied to the trip documents: (Q1) Search trips passing Shangtanglu street; (Q2) Search trips passing Shangtanglu AND Zhonghegaojia streets; (Q3) Search trips passing Shangtanglu OR Zhonghegaojia streets. Table 2 is the query performance on the trip documents for one day, one week, and one month period. It shows that the query time on the whole month data is very fast at 1.75 seconds for 590k hits (Q1), 1.32 seconds for 141k hits (Q2), and 2.85 seconds for nearly 0.9 million (918k) hits (Q3). These times are proportional to the number of hits. When users query one day (or one week), which are the cases for most analysis tasks, the queries can be done within 0.5 seconds. The indexing is fast, e.g., one month data is completed in around 22 minutes. Here the size of the index file is 6.55GB, which is 21% smaller than 8.5GB of the raw trip documents. This is achieved by the data compression of the engine.

We further conduct Q1-Q3 queries on the trajectory documents. The performance is shown in Table 3 for different time periods. Please note that we use each index file for a ten-minute period as discussed before. The query performance is fast, at less than 1 second for one day/week data. For one month data that has 23GB index files, the queries can be done in less than 5 seconds (i.e., 4.69 seconds for Q3 with nearly 2 million hits). The index size of 23GB is about 40% smaller than the raw trajectory documents at 38GB due to the compression.

Table 4 shows the performance of proximity and fuzzy queries. A proximity search *"shangtanglu zhonghegaojia"∼10* and a fuzzy search *shangtanglu∼0.8* are applied over trip documents for one day, one week, and one month data, respectively. The table shows that these queries are completed with fast performance too.

Comparison with Region Query In our approach, users find trajectories or trips by simply giving street/POI names or speed descriptions. It is different from the conventional method which provides spatial regions to find trajectories. The region queries are usually built up on spatial databases where the hierarchical structures, (e.g., R-trees, B-trees, K-d tree) of spatial cells. Please note that the region query model cannot easily implement queries by street names. It needs to conduct geospatial region search and filter the results according to the geometry of the streets. For example, an approximate method [31] used a set of small cells on the surface of a street to support users brushing a street for querying. Each cell was handled as a region to complete the query. This approach cannot easily handle name queries, especially for search conditions involving multiple streets. On the other hand, our model also cannot directly answer queries given a geospatial region, since extra computing is needed to find the streets inside the region. In the situations that users do not know street names, the map-based query is still needed. In summary, the two query models are *complementary* to each other, which can be combined to fulfill various visual query tasks of trajectories.

We compare our performance with the existing visual system of taxi data by Wang et al. [31]. Following their method, we first find a set of regions on the surface of the street, Shangtanglu, used in Q1 above. Then we perform region queries using their method which combines spatial trees with hash tables for indexing trajectories. Wang's system uses 712 seconds to create indexing from trip files of one week. Then Q1 costs 0.53 seconds. In comparison, Lucene is faster using around 336 seconds for indexing and 0.21 seconds for the query. Note that Wang's system cannot handle the whole month data in one workstation, because it uses a great amount of memory in creating spatial data structures. Moreover, its region query does not support searches with two street names by logic operations, such as Q2 and Q3. Extra join and filter algorithms are needed.

Table 3. Query performance on trajectory documents.

Time Period	No. of Files	Index Size	Indexing Time (sec)	Q1 Time (sec)	Q1 Hits	Q2 Time (sec)	Q2 Hits	Q3 Time (sec)	Q3 Hits
One Day	144	794MB	77	0.17	38k	0.21	3k	0.28	63k
One Week	1008	4.98GB	840	0.89	257k	0.65	19k	1.0	427k
One Month	4464	23GB	3,951	3.66	1,169k	2.14	84k	4.69	1,921k

Table 4. Query performance on proximity and fuzzy queries.

Time Period	Proximity Query		Fuzzy Query	
	No. of Hits	Time (sec)	No. of Hits	Time (sec)
One Day	5k	0.24	27k	0.25
One Week	16k	0.29	96k	0.36
One Month	118k	0.69	630k	1.03

7 VISUALIZATION SYSTEM OF SEMANTICTRAJ

7.1 Design Rationale

Trajectory data dynamically evolves over geospatial-temporal dimensions. Users exploring the data conduct interactions over both spatial and time dimensions. In the introduction we have exemplified complex interactions people need to conduct for different tasks. The major goal of SemanticTraj is to facilitate an easier and more intuitive way for users interacting with trajectory data, which complements to map-based visualizations and interactions. Therefore, the design of SemanticTraj focuses on

- Helping users easily externalize their ideas of data queries: In SemanticTraj, the visual interface allows direct input of semantic names and terms as in a familiar text search interaction.

- Promoting prompt understanding of query results: The query results are shown on a map with geographic context with zooming and panning operations. Visually studying them, such as finding what streets they traversed, is often confounded by cluttering while drawing many trajectories on road network. Semantic information, including text labels and meta-summaries, is then added to the visualization. The reason is that text is a very effective tool to enhance user understanding. Text labels are used to show important information over the map, and meta-summaries are used to automatically describe the behavior of individual trips or a group of trips.

- Guiding users in data explorations with easy interactions: SemanticTraj helps users discover insights with guided data refinement and drilling down. A table view allows users to look through all retrieved trajectories and select individual ones for examination. Scatterplots, parallel coordinates, and parallel sets are presented for users to find interesting results. Users can interact with meta-summaries to further drill down according to interesting streets.

7.2 Visual Exploration of Taxi Trajectories

Fig. 3 shows the visualization interface of SemanticTraj. It consists of widgets for query construction and showing a set of coordinated views. These views are synchronized when users make selection or filtering on each of them. Please watch a supplemental video of the visual system. Next, we describe how the visual system facilitates effective and efficient visual exploration of taxi trajectories. We use an example of querying taxi trips passing Shangtanglu street in the morning (7am-9am) of Dec 6, 2011.

7.2.1 Inputting Semantic Query Sentence

Fig. 3(1) is the input box of a semantic query. Users can write a textual search sentence with name and time conditions. Auto-complete provides search suggestions with similar names. Lucene's query parser will process the input. For the example task, users input *Shangtanglu AND PickUp:[07:00:00 TO 09:00:00]*, where the AND combines the condition of street name and the range query of pick-up time.

Users can alternatively open a form and fill in it to create a query sentence. The sentence is automatically generated from the input fields. Fig. 5 shows the form where users choose the time period, fill in two street names, and select the AND operation. The auto-generated query sentence is displayed in the input box. Proximity and fuzzy queries can be generated in a similar way.

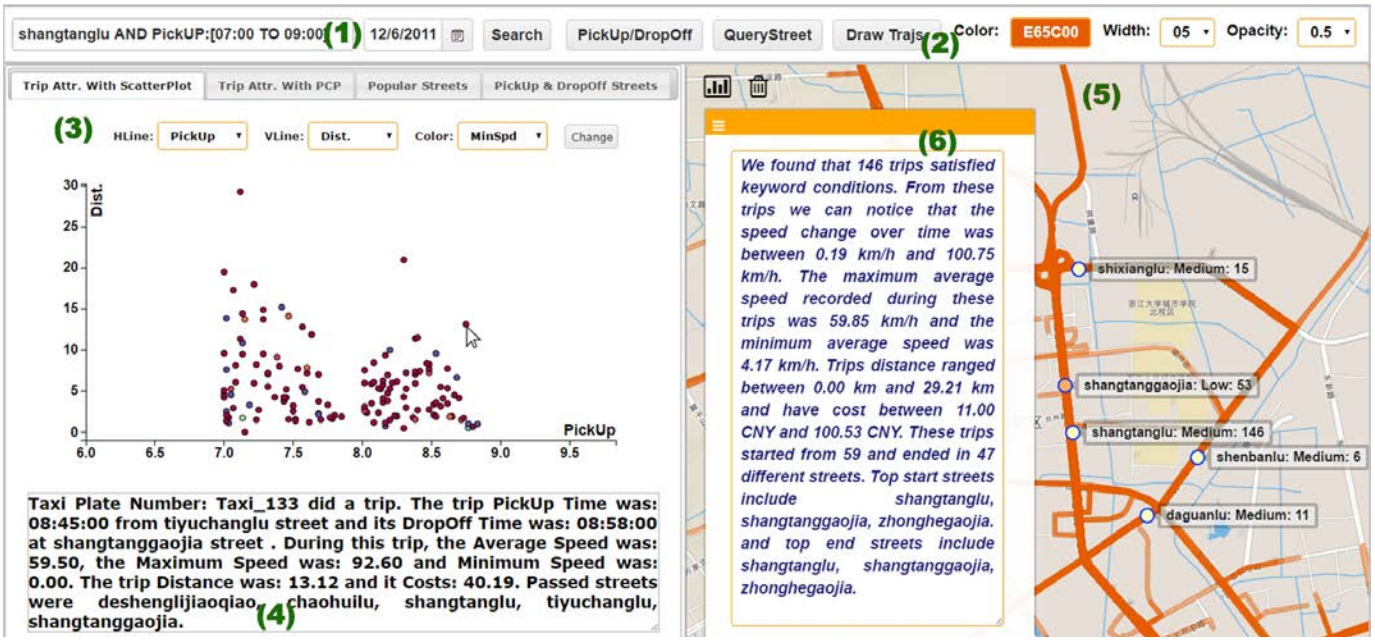


Fig. 3. *Using SemanticTraj to visualize taxi trips which passed Shangtanglu street of Hangzhou, China in the morning (7am-9am) of Dec 6, 2011. See details in Sec. 7. (1) Query input box accepting semantic query conditions as Shangtanglu AND PickUp:[7:00-9:00]; (2) Visualization control panel for adjusting the appearance; (3) Scatterplot view for users to study search results. Other visual tools can be selected in this view; (4) Meta-summary of a selected trip which automatically summarizes the trip fact; (5) Map view showing trip trajectories. Text labels are displayed on critical streets about its role in these trips; (6) A meta-summary of the group of all 146 result trips. Users can interact with the name tags to filter trips.*

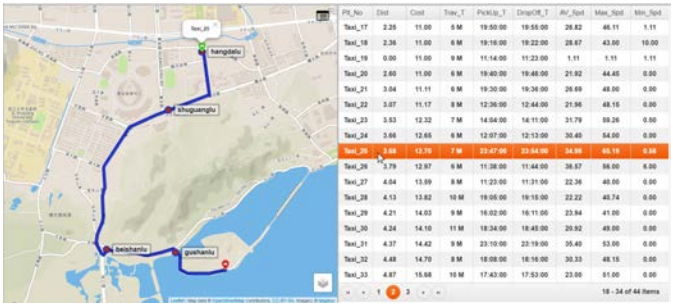


Fig. 4. *Using a table view to study individual taxi trajectories.*

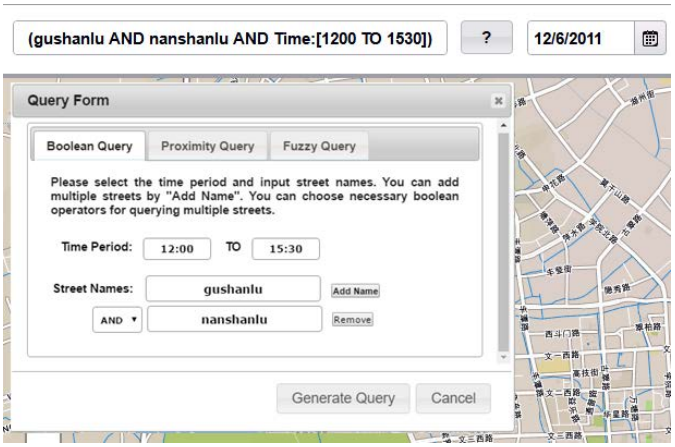


Fig. 5. *Users can fill in a form to automatically generate query sentence in the input box.*

7.2.2 Displaying Query Results with Semantic Information

Map View: Fig. 3(5) is a zoom-in view of the acquired taxi trips on the map. The paths of these trips are drawn with the same opacity, so that the color density on a road reflects how often it is used by these trips. Users can change the drawing attributes such as color, line width, or opacity for their favorite appearance. They can also show the pick-up and drop-off locations as map markers. These display options are controlled by the panel shown in Fig. 3(2).

Text Labels: Text labels are given on a few streets, which show the street names, the speeds of the trips, and the taxi counts on the streets. For example, the topmost label in Fig. 3(5) is "shixianglu: medium: 15". It indicates 15 taxi pick-ups happened on Shixianglu street and the average travel speed on this street is medium. The system provides a variety of labeling options to show the top streets where pick-ups (or drop-offs) happen, or the top streets with a fast (slow) speed. The number of labels is controllable by users to reduce clutter. They can toggle on/off different items (name, speed, etc.) in the labels. Users can also click on a street to show its label.

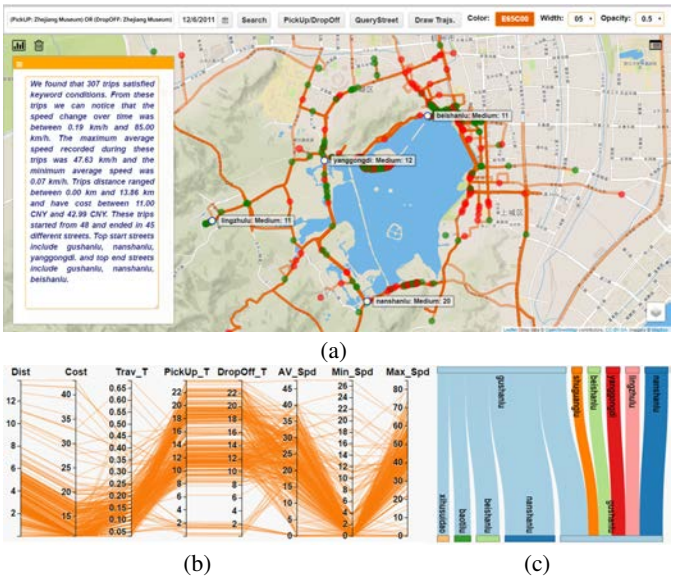
Meta-summary: Fig. 3(6) displays the meta-summary of the taxi trips in textual description. It summarizes the total number of trips, the speed information of this group, and the popular streets. From this report, users can get an idea of the retrieved taxi trips immediately. They can also click on a name to refine the trips passing the corresponding street. The summary can be toggled on and off by users.

7.2.3 Interactively Refining Query Results

Table View Users can study the attributes of all the result trips in a table shown in Fig. 4. It displays the plate number (anonymized for privacy protection), trip distance (length of the path), trip cost (computed by the taxi fare rule of Hangzhou), travel time, and pick-up and drop-off streets. The table also shows the max, min, and average speed of each trip. The trips can be ordered using different scoring functions. Users can choose single or multiple trips from the table. Selected trips will be displayed on the map (Fig. 4), while the major streets are labeled so that users can efficiently study the paths.

Scatterplot View Fig. 3(3) is a scatterplot which draws each taxi trip as a dot. Users can choose the attributes mapped to the axes and the color of dots. The attributes are trip distance, cost, traverse time, pick-up time, drop-off time, max speed, min speed, and average speed. Users can brush and choose a subset of trips for further study by selecting a set of dots. Moreover, users can hover over a dot to show its individual meta-summary. The behavior of this trip is displayed in a descriptive sentence (Fig. 3(4)).

Pick-up and Drop-off Relations A parallel sets view [19] shows the



(a)

(b) (c)

Fig. 6. *Provide shuttle buses for tourists of Zhejiang Museum after query taxi trips leaving or arriving at the museum. (a) Taxi trips are shown where street names with frequent pick-ups (potential bus stops) are easily found by the text labels; (b) The PCP view of the trips show their attributes. (c) The parallel sets view reveals the pick-up and drop-off relations.*

relationships of the pick-up and drop-off streets (Fig. 6c). The top streets used by pick-ups are drawn on the top horizontal line and the top streets used by drop-offs are drawn on the bottom. The width of ribbons linking two streets is proportional to the number of trips traveling between them. A ribbon's color is the same as the color of pick-up streets. Fig. 6c shows that the trips starting from Gushanlu end on a variety of streets. Users can click on a ribbon to investigate the trips it represents in other views.

Parallel Coordinates Plot A parallel coordinates plot (PCP) helps users identify trends as well as outliers of the taxi trips. For example, Fig. 6b shows that most taxi pick-ups/drop-offs happen between 8am and 10pm. Users can hover over a line to highlight a trip on the map and see the meta-summary of the trip.

Circular Heatmaps Circular heatmaps (Fig. 9) visualizes the statistical data such as the average speed of streets (see Sec. 8.4). The whole day is divided into 24 hours and each hour is represented by 6 arcs so that each unit reflects the average speed (or other attributes) in a 10-minute period.

We use a qualitative color spectrum from ColorBrewer [18]. More spectrums can be selected such as colorblind-safe palettes.

8 Usage Scenarios

We show the usability of our prototype in a set of illustrative usage scenarios of taxi data visual analytics.

8.1 Provide shuttle buses for tourists

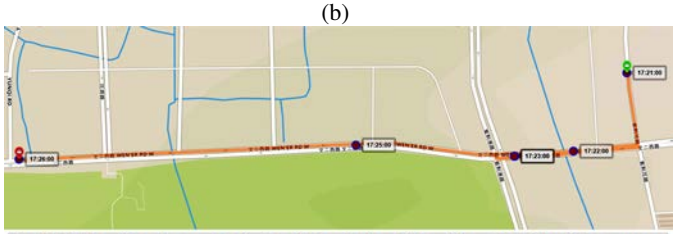
Zhejiang Museum wanted to improve their service to customers. The museum clerks used SemanticTraj to query taxi trips with their own POI name (i.e., Task 3). By inspecting the results, they found the paths, popular pick-up and drop-off locations, and when passengers took taxis to reach the museum. The museum would provide complementary shuttle buses based on the results. In particular, the clerks formed the query input *Pick-up:Zhejiang Museum OR Drop-off:Zhejiang Museum* over a week from Dec 5 to Dec 11. Fig. 6a showed the results on the map where the drop-off/pick-up locations were shown in green/red dots. The visualization provided a general overview of their distribution, but the clerks need more information to decide the locations of candidate bus stations. They looked at the meta-summary, where basic facts of the 329 taxi trips were summarized. They toggled on the text labels for the top pick-up streets (Nanshanlu, Beishanlu, Lingzhulu and Yanggongdi). The street names and pick-up counts were displayed

(a)

Plt_No	Dist	Cost	Trav_T	PickUp_T	DropOff_T	AV_Spd	Max_Spd	Min_Spd
Taxi_0	1.98	11.00	4 M	17:22:00	17:26:00	31.00	57.00	0.00
Taxi_1	2.48	11.00	5 M	17:24:00	17:29:00	28.40	62.97	0.00
Taxi_2	2.73	11.00	5 M	17:21:00	17:26:00	33.83	67.00	0.00
Taxi_3	2.53	11.00	2 M	17:24:00	17:26:00	31.56	59.00	0.00
Taxi_4	1.82	11.00	5 M	17:22:00	17:27:00	14.82	51.86	0.00
Taxi_5	2.73	11.00	4 M	17:17:00	17:21:00	35.81	59.26	0.00
Taxi_6	3.13	11.33	5 M	17:18:00	17:23:00	25.00	59.26	0.00
Taxi_7	3.77	12.92	10 M	17:19:00	17:29:00	12.76	60.00	0.00
Taxi_8	4.21	14.03	4 M	17:25:00	17:29:00	13.29	49.00	0.00

1 - 9 of 9 items

(b)



Taxi Plate Number: Taxi_2 did a trip. The trip PickUp Time was: 17:21:00 from zijinghualu street and its DropOff Time was: 17:26:00 at wenerxilu street . During this trip, the Average Speed was: 33.83, the Maximum Speed was: 67.00 and Minimum Speed was: 0.00. The trip Distance was: 2.73 and it Costs: 11.00. Passed streets were wenerxilu, zijinghualu.

(c)

Fig. 7. *Find a criminal suspect passenger on taxi trips. (b) All result trips shown on the map; (b) A list of the trips on the table; (c) One of the two suspicious trips.*

on the map. These streets were good candidates. Users clicked on the meta-summary and labels to filter trips passing a specific street.

The clerks further filtered the result trips in the morning and afternoon (not shown here). They found that Nanshanlu and Beishanlu were used for pick-up and drop-off in the morning, but in the afternoon Beishanlu was mostly used only as drop-off locations. This fact helped them schedule the buses. Moreover, there was a far-away suburban visitor center of Lingyin mountain at Lingzhulu. The system showed that most passengers took taxis from it to the museum in the morning. So shuttle buses could be arranged there in the morning. The PCP in Fig. 6b allowed the clerks study attributes of the trips. For example, the trips with a pick-up time of (14-17) were less than that of (10-14) and (17-19), which can be used to decide shuttle bus schedules.

8.2 Find criminal suspect taxi passenger

A crime happened in Hangzhou at around 5pm, Dec, 5. The characteristics of the suspect was described by the victim and then broadcasted on TV. One eyewitness reported to the police that he saw the suspect on a taxi traveling at Zijinghualu Street at around 5:10-5:30pm, and another citizen said that he witnessed the taxi at Wenerxilu Street at around 5:20-5:30pm. To find potential taxi trips of the suspect, a police officer queried the trip documents by the two street names (i.e., Task 2) and the time periods. The query condition was {*Zijinghualu*

Fig. 8. *Identify taxi drivers' abnormal behavior.*

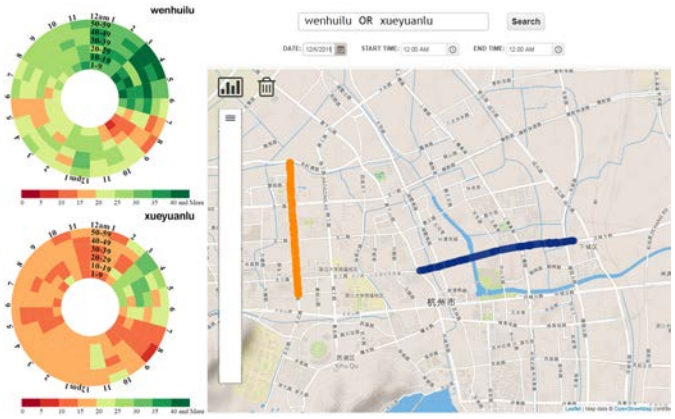


Fig. 9. *Assess traffic situations of two streets.*

AND Wenerxilu AND {*Pick-up Time:[17:00:00 TO 17:30:00]*} *AND* {*Drop-off Time* > *17:20:00*}}. Please note the times given by the witnesses were not accurate so the query covered a larger time range. Fig. 7a showed the acquired trips on the map, where their pick-up and drop-off locations were shown as markers. The officer also marked the positions where the witnesses were and the criminal site. Fig. 7b showed the list of these trips on the table view. The officer browsed the list and interactively clicked on each trip to inspect its path on the map. The officer quickly identified two trips of Taxi_2 and Taxi_3 which passed the site of the witnesses. Fig. 7c showed one of the two suspicious trips. The meta-summary described its travel behavior, including pick-up/drop-off streets, passed streets, and speeds. The information allowed the officer to further investigate the trip.

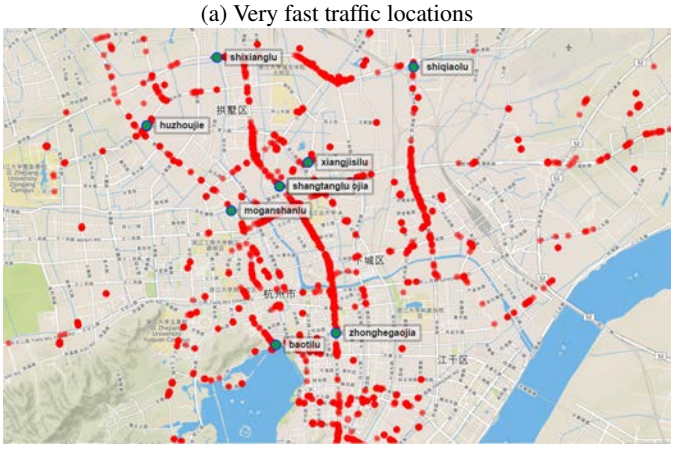
8.3 Identify taxi drivers' abnormal behavior

A taxi company manager wanted to find drivers who had improper driving performance in a day. The manager queried the trajectory documents to find taxis having an abrupt change of speed (i.e., Task 5). She observed Fig. 8 which showed the results using the query condition {{*Slow VeryFast* ∼*1*} *in DSpeed*}, which implied speeding since such a quick acceleration from less than 20 km/h to more than 100 km/h was not proper in the city center. The taxi IDs and the streets related to these abnormal trajectories were labeled on the map.

8.4 Study traffic information over streets

The residents of Hangzhou wanted to learn the traffic information of the city. In Fig. 9, a resident queried the trajectory documents of a day with the names of two street close to his home, namely Wenhuilu (blue) and Xueyuanlu (orange). The two streets were highlighted on the map. Two circular heatmaps showed the average speed of them throughout the day. The resident compared the two heatmaps. She found that both of them had rush hours between 7am and 9am. She also found that Xueyuanlu was much busier than Wenhuilu in most time periods even in midnight between 12am and 3am.

(a) Very fast traffic locations



(b) Traffic jam locations

Fig. 10. *Assess city-wide traffic information.*

In another scenario, a practitioner in the Bureau of Transportation wanted to assess urban traffic situations. He queried the trajectory documents by {{*VeryFast VeryFast*} *in DSpeed*}. Fig. 10a showed the result locations, where taxis drove very fast were indicated in red color. They manifested many major roads in Hangzhou. However, many parts of these roads had intermittent red dots, which indicated slow traffic.

He queried again with the query condition using ten continuous Slows as {{*Slow Slow ... Slow*} *in DSpeed*}. This query sentence is easily created in the proximity query on the form (Fig. 5). Fig. 10b showed the result locations, where a traffic jam possibly happened since the retrieved trajectories had consecutive slow speeds. Visual labels in the two figures helped the practitioner quickly identify street names.

8.5 Discussion: Benefits of Using Our System

In the illustrative usage scenarios, users are diversified including police officers, museum clerks, city residents, and taxi company managers. Our system is a good choice in completing their work for the following reasons: (1) It is very convenient for these general users to conduct searches by simply filling in street/POI names (or other descriptive terms). It would cost them extra effort to locate and select (e.g. by brushing) the streets over a map. (2) Our system works fast in giving results through a variety of visualizations. (3) The summaries and labels are read quickly for semantic knowledge.

9 Evaluation

9.1 Expert Feedback

We conducted in-person interviews with two domain experts: one is an urban transportation researcher and the other one is a criminologist. Both of them hold a PhD degree in their fields. First, we introduced our method. Then we demonstrated our system with several examples usage scenarios. Then, the experts used our system for interactive

exploration. Finally, they were provided a document describing our methodology and system. A few days later, the experts fed back to us with written documents of their opinions.

The criminologist has been working with police departments on fighting urban crime. She provided the following feedback: "Visual query using text search engine is an innovation for policing strategy. Street or POI names are more straightforward to criminologists than abstract geometry or spatial coordinates. This tool is *very useful because it is closer to our understanding of the real place featured with names instead of a virtual location represented by numbers*. With great volumes of trajectories data by virtue of everyday technology, the sharing and utilization of these massive data have presented great challenges for research and application in crime mapping. Moreover, the concepts and technologies of Web 2.0 represent demands for user-oriented interaction and collaboration. Most criminologists are not geographers or geographic information scientists. Hence, there is a huge gap between spatial technology advances and policing needs. Most current crime mapping tools rely on spatial query through geometric regions or buffers on a map to search trajectories, which is not as intuitive and easy as visual query for domain scientists and common users. This new tool addresses an emerging need to provide non-technical users with an integrated platform for spatial exploration and visualization. This practice will open up a rich empirical context for interdisciplinary studies and policy interventions."

The urban transportation researcher provided the following feedback: "As a former urban planner and currently an urban geographer, I have spent years of efforts on understanding the vehicle flow on the urban street network and its uneven traffic distribution across space and over time. Many methods have been tried, ranging from surveys and field observation to spatial statistical analysis. However, these methods do not recognize the difficulty of big transportation data and impossibility of visualizing such data using conventional approaches. It warrants notice that spatial distribution of real traffic flow in the street network may vary from time to time, which poses a challenge to store and query the data for real-time use. The developed tool can solve two major issues: store and query the increasing data size especially real-time feature; analyze and identify the network structure of transportation flow in a fast and visual platform. This tool is a good example of spatial intelligence, which allows efficient data integration for large-volume and near real time spatial and non-spatial data (multi-source data). In addition *it allows users who have very few spatial data handling skills to conduct space-time data analysis easily and effectively*. Hence, this tool can serve as a practice, research and education platform by delivering geographic information service in the data-rich age which is featured by the unprecedented terabytes of digitized data. "

They suggested that we improve the work mostly by making the web-based system remotely operated by multiple users in real-time, and providing more interactions for users to find more city data. These comments provide several directions of our future work.

9.2 User Study of Visual System

We performed a preliminary user study with 15 CS graduate students (6 females and 9 males with ages from 23 to 34). The goals of the study were to evaluate whether the visual system is easy to use and whether the semantic meta summaries and labels, is helpful in visual analytics. The subjects conducted the study one by one. First, the subjects were given an introduction describing the system and the tasks. Then, they practiced on the system for 5 minutes. After that, they were asked to search for a street, Gushanlu, to find all trips passing it using a name query. Next, they were asked to write down the answers to the following questions: "How many trips passed this street?; "What is the maximum average speed?; "What is the top two pick-up streets?; "What is the maximum fare cost?; and "How many trips start from (nanshanlu) street?. Each subject conducted the above task in two sequential sessions. Group 1 include half of all subjects (picked randomly) who answered the questions using the full system (S1) in the first session. Then they answered the same questions again on new answer sheets using the system without meta-summaries and labels (S2) in the second session. Group 2 including another half of the subjects used S2 in the first session and S1 in the second session. At the end of the study, they answered the following questions: (A) Overall, do you think the system can help complete such tasks intuitively? (B) To complete such tasks, which visual tools were more useful (Meta-summary, visual labels, or other tools)? (C) To what extent the visual system is easy to use: Very Easy, Easy, Fair, or Poor? and (D) What is your suggestion for improvement?

The average task completion time using S1 and S2 were 2 minutes and 5 minutes, respectively. 93% of subjects achieved correct answers compared to the ground truth. There was no significant difference on the completion time and correctness between Group 1 and Group 2. The results indicated that the meta-summaries and the labels were helpful in accelerating the visual analytics process. For question (A), all subjects agreed that the system was qualified in completing the tasks. For (B), 67% of the subjects preferred to use Meta-summary and Labels to complete the tasks. 33% of the subjects preferred to use other visual representations (e.g., PCP and scatterplot). For (C), 73% of the subjects agreed that this system was very easy to use and 27% of the subjects said it was easy. In addition, the subjects were very excited about our prototype. They suggested many ideas for improving our prototype, such as showing more interesting information that reflects the semantic of trips on the map; using the system for real-time data; etc. This study showed that our prototype system is easy to use and can be of great interest for urban trajectory study.

10 Conclusion and Discussion

We have proposed a new approach to interacting with massive taxi trajectory data sets. It utilizes textualization and taxi documents so that the interactions can be applied through semantic rich operations. In this paper, semantic information refers to the "meaning in language", where we map the numeric values of GPS points and speeds to names and descriptions in natural language for easy query, visualization and user understanding. Semantics may also imply high-level, summarized information describing the pattern and knowledge hidden in languages. We will further investigate in this direction where text mining tools can be used to analyze the taxi documents for deeper insights. The underlying text search engine provides a new way of data management and fast query support for various Boolean conditions, which is complementary to existing region queries. General users are thus provided easy, intuitive tools with enhanced guidance in their visual exploration for a set of analytical tasks. We developed a prototype visual analytics system with a set of visualization tools for users to conduct interactive visual exploration. In the future, we will enhance the system by improving the visual interface and disseminate the system to domain users.

Acknowledgments

We thank the reviewers for their highly constructive feedback and suggestions. This work is partially supported by US NSF grants 1535031, 1535081, and 1416509. Wei Chen is supported by 973 Program of China (2015CB352503), NSFC(61232012,61422211).

References

[1] N. Andrienko and G. Andrienko. Spatial generalization and aggregation of massive movement data. *Visualization and Computer Graphics, IEEE Transactions on*, 17(2):205–219, 2011.

[2] N. Andrienko and G. Andrienko. Visual analytics of movement: An overview of methods, tools and procedures. *Information Visualization*, 12(1):3–24, 2013.

[3] Apache. Apache lucene 4.0.0 documentation. https://lucene.apache.org/core/4_0_0/, 2016.

[4] A. Bialecki, R. Muir, G. Ingersoll, and L. Imagination. Apache lucene 4. In *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*, 2012.

[5] V. Bogorny, C. Renso, A. R. de Aquino, F. de Lucca Siqueira, and L. O. Alvares. Constant a conceptual data model for semantic trajectories of moving objects. *Transactions in GIS*, 18(1):66–88, 2014. doi: 10.1111/tgis.12011

[6] M. Borkin, Z. Bylinskii, N. Kim, C. Bainbridge, C. Yeh, D. Borkin, H. Pfister, and A. Oliva. Beyond memorability: Visualization recognition

and recall. *Visualization and Computer Graphics, IEEE Transactions on*, 22(1):519–528, 2016. doi: 10.1109/TVCG.2015.2467732

[7] S. Büttcher, C. Clarke, and G. V. Cormack. *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press, 2010.

[8] P. S. Castro, D. Zhang, C. Chen, S. Li, and G. Pan. From taxi gps traces to social and community dynamics: A survey. *ACM Comput. Surv.*, 46(2):17:1–17:34, Dec. 2013. doi: 10.1145/2543581.2543584

[9] V. P. Chakka, A. C. Everspaugh, and J. M. Patel. Indexing large trajectory data sets with seti. *Ann Arbor*, 1001:48109–2122, 2003.

[10] Y. Chen, S. Barlowe, and J. Yang. Click2annotate: Automated insight externalization with rich semantics. In *Proceedings of IEEE VAST*, pp. 155–162, 2010.

[11] D. Chu, D. A. Sheets, Y. Zhao, Y. Wu, J. Yang, M. Zheng, and G. Chen. Visualizing hidden themes of taxi movement with semantic transformation. In *Pacific Visualization Symposium (PacificVis), 2014 IEEE*, pp. 137–144. IEEE, 2014.

[12] P. Cudre-Mauroux, E. Wu, and S. Madden. Trajstore: An adaptive storage system for very large trajectory data sets. In *IEEE International Conference on Data Engineering (ICDE)*, pp. 109–120. IEEE, 2010.

[13] J. Dai, B. Yang, C. Guo, and Z. Ding. Personalized route recommendation using big trajectory data. In *Data Engineering (ICDE)*, pp. 543–554. IEEE, 2015.

[14] K. Deng, K. Xie, K. Zheng, and X. Zhou. Trajectory indexing and retrieval. In *Computing with Spatial Trajectories*, pp. 35–60. Springer, 2011.

[15] N. Ferreira, J. Poco, H. T. Vo, J. Freire, and C. T. Silva. Visual exploration of big spatio-temporal urban data: A study of new york city taxi trips. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2149–2158, Dec. 2013. doi: 10.1109/TVCG.2013.226

[16] R. Fileto, C. May, C. Renso, N. Pelekis, D. Klein, and Y. Theodoridis. The baquara^2 knowledge-based framework for semantic enrichment and analysis of movement data. *Data Knowl. Eng.*, 98:104–122, 2015. doi: 10.1016/j.datak.2015.07.010

[17] A. Guttman. R-trees: A dynamic index structure for spatial searching. *SIGMOD Rec.*, 14(2):47–57, June 1984. doi: 10.1145/971697.602266

[18] M. A. Harrower and C. A. Brewer. ColorBrewer.org: An Online Tool for Selecting Color Schemes for Maps. *The Cartographic Journal*, 40(1):27–37, 2003.

[19] R. Kosara, F. Bendix, and H. Hauser. Parallel sets: Interactive exploration and visual analysis of categorical data. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):558–568, 2006. doi: 10.1109/TVCG.2006.76

[20] R. Krüger, D. Thom, M. Wörner, H. Bosch, and T. Ertl. Trajectorylenses - a set-based filtering and exploration technique for long-term trajectory data. *Computer Graphics Forum*, 32:451–460, 2013.

[21] N. Lathia and L. Capra. Mining mobility data to minimise travellers' spending on public transport. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pp. 1181–1189. ACM, New York, NY, USA, 2011. doi: 10.1145/2020408.2020590

[22] H. Liu, Y. Gao, L. Lu, S. Liu, L. Ni, and H. Qu. Visual analysis of route diversity. *IEEE Conference on VAST*, pp. 171–180, 2011.

[23] S. Liu, W. Cui, Y. Wu, and M. Liu. A survey on information visualization: Recent advances and challenges. *Vis. Comput.*, 30(12):1373–1393, Dec. 2014. doi: 10.1007/s00371-013-0892-3

[24] M. Lu, Z. Wang, and X. Yuan. Trajrank: Exploring travel behaviour on a route by trajectory ranking. *Proceedings of IEEE Pacific Visualization Symposium*, pp. 14–17, 2015.

[25] B. C. Ooi, R. Sacks-Davis, and J. Han. Indexing in spatial databases. http://www.comp.nus.edu.sg/ ooibc/spatialsurvey.pdf, 1993.

[26] B. Pan, Y. Zheng, D. Wilkie, and C. Shahabi. Crowd sensing of traffic anomalies based on human mobility and social media. In *Proceedings of SIGSPATIAL'13*, pp. 344–353. ACM, New York, NY, USA, 2013. doi: 10.1145/2525314.2525343

[27] C. Parent, S. Spaccapietra, C. Renso, G. Andrienko, N. Andrienko, V. Bogorny, M. L. Damiani, A. Gkoulalas-Divanis, J. Macedo, N. Pelekis, Y. Theodoridis, and Z. Yan. Semantic trajectories modeling and analysis. *ACM Comput. Surv.*, 45(4):42:1–42:32, Aug. 2013.

[28] J. Pu, S. Liu, Y. Ding, H. Qu, and L. Ni. T-watcher: A new visual analytic system for effective traffic surveillance. In *Proceedings of Mobile Data Management*, pp. 127–136, 2013.

[29] S. Rinzivillo, S. Mainardi, F. Pezzoni, M. Coscia, D. Pedreschi, and F. Giannotti. Discovering the geographical borders of human mobility. *KI - Knstliche Intelligenz*, 26:253–260, 2012.

[30] H. Su, K. Zheng, K. Zeng, J. Huang, S. Sadiq, N. J. Yuan, and X. Zhou. Making sense of trajectory data: A partition-and-summarization approach. In *IEEE Data Engineering (ICDE)*, pp. 963–974. IEEE, 2015.

[31] F. Wang, W. Chen, F. Wu, Y. Zhao, H. Hong, T. Gu, L. Wang, R. Liang, and H. Bao. Visual reasoning approach for data-driven transport assessment on urban road. In *IEEE Conference on Visual Analytics Science and Technology*, pp. 103–112. IEEE, Oct. 2014.

[32] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. van de Wetering. Visual traffic jam analysis based on trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2159–2168, 2013. doi: 10.1109/TVCG.2013.228

[33] Z. Wang, T. Ye, M. Lu, X. Yuan, H. Qu, J. Yuan, and Q. Wu. Visual exploration of sparse traffic trajectory data. *IEEE Trans. on Vis. Comp. Graph.*, 20(12):1813 – 1822, 2014.

[34] J. Yuan, Y. Zheng, X. Xie, and G. Sun. T-drive: Enhancing driving directions with taxi drivers' intelligence. *IEEE Trans. on Knowl. and Data Eng.*, 25(1):220–232, Jan. 2013. doi: 10.1109/TKDE.2011.200

[35] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang. T-drive: Driving directions based on taxi trajectories. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '10, pp. 99–108. ACM, New York, NY, USA, 2010. doi: 10.1145/1869790.1869807

[36] N. J. Yuan, Y. Zheng, L. Zhang, and X. Xie. T-finder: A recommender system for finding passengers and vacant taxis. *IEEE Trans. on Knowl. and Data Eng.*, 25(10):2390–2403, Oct. 2013. doi: 10.1109/TKDE.2012.153

[37] F. Zhang, D. Wilkie, Y. Zheng, and X. Xie. Sensing the pulse of urban refueling behavior. In *Proceedings of UbiComp '13*, pp. 13–22. ACM, New York, NY, USA, 2013. doi: 10.1145/2493432.2493448

[38] B. Zheng, N. J. Yuan, K. Zheng, X. Xie, S. Sadiq, and X. Zhou. Approximate keyword search in semantic trajectory database. In *IEEE Data Engineering (ICDE)*, pp. 975–986. IEEE, 2015.

[39] K. Zheng, S. Shang, N. J. Yuan, and Y. Yang. Towards efficient search for activity trajectories. In *IEEE Data Engineering (ICDE)*, pp. 230–241. IEEE Computer Society, 2013.

[40] Y. Zheng, L. Capra, O. Wolfson, and H. Yang. Urban computing: Concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology*, 2014.

[41] Y. Zheng, Y. Liu, J. Yuan, and X. Xie. Urban computing with taxicabs. In *Proceedings of the 13th International Conference on Ubiquitous Computing*, UbiComp '11, pp. 89–98. ACM, New York, NY, USA, 2011. doi: 10.1145/2030112.2030126

[42] Y. Zheng and X. Zhou. *Computing with Spatial Trajectories*. Springer, 2011.

[43] J. Zhou, A. K. Tung, W. Wu, and W. S. Ng. R2-d2: a system to support probabilistic path prediction in dynamic environments via semi-lazy learning. *Proceedings of the VLDB Endowment*, 6(12):1366–1369, 2013.

[44] J. Zimmerman, A. Tomasic, C. Garrod, D. Yoo, C. Hiruncharoenvate, R. Aziz, N. R. Thiruvengadam, Y. Huang, and A. Steinfeld. Field trial of tiramisu: Crowd-sourcing bus arrival times to spur co-design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pp. 1677–1686. ACM, New York, NY, USA, 2011. doi: 10.1145/1978942.1979187