Towards Building Smart Maps from Heterogeneous Data Sources

Faizan Ur Rehman*, Imad Afyouni†, Ahmed Lbath‡ and Saleh Basalamah§
*‡ LIG, University of Grenoble Alpes, France
* Science and Techology Unit, Umm Al-Qura University, Saudi Arabia
† § Technology Innovation Center, Wadi Makkah, Umm Al-Qura University, Saudi Arabia
§ College of Computer Science, Umm Al-Qura University, Saudi Arabia
Email: *fsrehman@uqu.edu.sa, †iafyouni@gistic.org, ‡ahmed.lbath@imag.fr, §sbasalamah@uqu.edu.sa

***Abstract*—In today's busy world, users and authorities require better services to achieve their daily activities and tasks in a smart way by using available resources in an optimized manner. The variety of available data sources, starting from crowdsourced data, open governmental data, and other online sources can provide users with smart tools to better manage their daily activities. However, collecting and integrating this multitude of overlapping data sources is a challenging task. Particularly, digital maps are being extensively used to browse and share information about points of interest, plan trips, and to find optimized paths. Within this context, there is a real opportunity to enrich traditional maps with different knowledge-based layers extracted from the variety of available data sources. This paper introduces the concept of "*smart maps*" by collecting, managing, and integrating heterogeneous data sources in order to infer relevant knowledge-based layers. Unlike conventional maps, *smart maps* extract live events (e.g., concert, competition, incident), online offers, and statistical analysis (e.g., dangerous areas) by encapsulating incoming semi- and unstructured data into structured generic *packets*. These packets are processed to extract statistical knowledge on accident-prone and safe areas, and detect *Events of Interest* (EoI), based on a multi-dimensional clustering technique. This approach lays the ground for delivering different intelligent services and applications, such as: 1) city explorer that provides latest information collected from multiple sources about places and events; and 2) route and trip planning that leverage smart map framework to recommend safe routes.**

***Keywords*–Smart Maps; Events of Interest; Social Networks; Crowdsourcing.**

I. INTRODUCTION

Publicly available data is increasing rapidly with a rate of 30% every year and will continuously keep growing with the advancement of technologies in sensors, smartphones and the Internet of Things. Data from multiple sources can improve the coverage for providing relevant knowledge about surrounding events and points of Interest (POIs) [1]. Moreover, adding social network data can provide fruitful insights and complementary sources of information. Social data can be used in various scenarios, such as emergency situation, inferring events in cities, and showing breaking news. Meanwhile, the use of digital maps is tremendously increasing with the aim of sharing preeminent information about current locations and spatial characteristics of surroundings. Big giants including *Google*, *Yahoo*, *MapQuest* and *Bing* generate dynamic layers about traffic updates, such as traffic jams, accidents, and congestions but they still lack to provide knowledge about statistical trends, ongoing events, and POI semantics and ranking from crowdsourced data. Such knowledge can be extracted and enriched from heterogeneous available data sources. Different knowledge-based layers can enrich existing traditional maps to

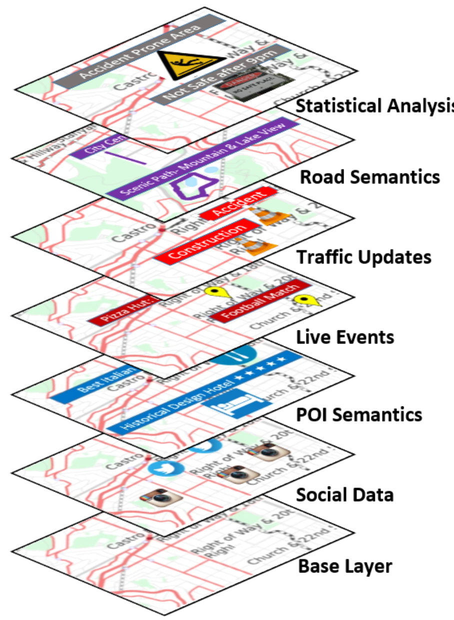

Figure 1. Dynamic layers of smart maps

know more about surroundings, and can be used to enhance existing spatio-temporal queries (see Fig. 1). In this work, we present a framework that collects and integrates data from different sources including crowdsourced data (social data including Twitter and Yelp), Open Street Map (OSM) data, and other online data (Google traffic, Open government). The system applies different mining and clustering techniques in order to convert existing digital maps into smart maps. Smart maps are maps that intelligently self-update themselves based on information extracted dynamically from heterogeneous data sources including social media streams, crowd-sourced data, sensors and online news sources. Smart maps can discover new content automatically by identifying new points of interests, events, or findings that were not specifically entered to the map. We believe that smart maps will form the next generation of digital maps by providing awareness about surroundings, such as events, traffic updates, road semantics (e.g., scenic or safe path), POI semantics (e.g., fast food restaurant), online offers, and statistical analysis (e.g., dangerous areas) as illustrated in Fig. 1.

Smart maps provide the following dynamic layers on top of existing maps: 1) *Events of Interest (EoI)*: live events in cities, such as concerts, football matches, jobs hiring and other relevant information (e.g., pizza hut discount, sales in CityMax) are displayed at different levels of abstraction; 2) *Statistical Analysis*: illustrates analysis of (un)safe areas by extracting knowledge on accident prone areas, safe or polluted areas; 3) *Traffic updates from social data*: shows information about traffic constraints, such as accidents or road blocks collected from social data; and 4) *POI Semantics*: describes semantics and ratings of geographical places from crowdsourced data (e.g., using Yelp data to judge quality of POIs, such as best Italian pizza, historical design hotel, etc.).

The contributions of this paper are manifold: a) collect, store and clean structured, semistructured or unstructured crowdsourced data. This includes digesting microblog social data (Twitter, Yelp), OSM, online data in real-time (Google Traffic APIs) and other historical open government data (Crime and accident data); b) design a common schema to resolve data conflicts and integration issues of social data, and to increase the conciseness and correctness of data; c) extracting relevant knowledge by applying state-of-the-art text mining techniques and correlation to find Events of Interest (EoI); d) visualize smart map layers in an interactive way, and f) taking leverage of *smart maps* to enhance spatio-temporal queries. The following two applications are demonstrated on top of the smart maps framework:

- City Explorer: provides latest information collected from multiple sources about historical places, touristic places, dining, ongoing and upcoming events, shops, news, live social feeds, weather updates, traffic updates, semantics of points of interest.
- Routing Service: recommends optimized trips and paths to users not only based on traditional spatial and temporal data analysis, but also by taking into account safe areas to avoid accident and crime prone places.

The remainder of this paper is as follows. Section II discusses the related work from different perspectives. Section III introduces an overview of our system architecture. Section IV highlights the implementation and map visualizations; while Section V draws conclusions and future challenges.

II. RELATED WORK

In this section, we review relevant work on data collection and integration from smart cities, social networks mining, and events of interest detection from heterogeneous sources. We also review the existing maps and their layers and explain how smart maps can enrich current state-of-the-art of knowledge-based layers.

A. Data Collection and Integration for Smart Cities

The growth in population within cities will have an impact on transportation, city infrastructure and the global economic growth, etc. Therefore, it is suggested to convert big cities into smart cities to provide better facilities to citizens [2]. Big giants, including IBM (Smarter PlanetSmarter Cities) [3] and Microsoft (CityNext) [4], have launched several projects around the world to enhance city infrastructures and human life style. Smart cities generate multiple types of data sources which lead to the problem of data collection, cleaning, and integration [5]. Data collection and integration has been performed in the past for many applications, such as 1) analyses and extraction of points of interest from multiple web sources [6]; and 2) finding traffic flow for intelligent transportation services by using Twitter [7]. Data integration for multiple sources requires schema mapping, comparison of string resemblance, data dependency, source authenticity, and duplicate detection [5]. With this context and in order to detect live event from social data, we exploit multiple types of data sources by wrapping incoming data streams into packets, that is, a generic structured form out of semistructured or unstructured data. Like the analogy of TCP protocol, each packet has a metadata header, containing source, location, time, type, potential event and event properties, and a payload, containing the actual content.

B. State-of-the-art mapping technologies

Today's maps are often crowd-sourced, and make use of *'Volunteered Geographic Information (VGI)'* [8], where users can enrich maps with their own information. Researchers, authorities, and industries generate thousands of interactive analytics every year to meet their social and economic needs [9]. In addition, 'Live Maps' now contain data that is updated in real-time. For example, live updates of bus schedules, traffic conditions, restaurant opening hours, and road accidents can be displayed on *Google Maps*, among others. With the wide spread of social networks, people start to post their own social contributions on live maps, such as Foursquare check-ins, Flickr images, tweets [10], Yelp reviews, and news RSS feed [11]. Moreover, natural language processing (NLP) techniques were embedded to extract spatially-referenced news from online newspapers and tweets [11]. However, Live Maps still lack intelligence in extracting knowledge about new events of interest to safe areas and other statistical analysis.

C. Events of Interest Detection from Crowdsourced Data

In the last few years, the use of social media is increasing rapidly all round the world. The number of social media users worldwide has grown to around 2.51 billion in 2017 [12]. Many real-world applications, such as for traffic updates, surveillance, and earthquake detection, continuously collect crowdsourced data including social network data to detect events of interest. Early events of interest detection can be helpful to society, users, and authorities to take proper action on time with respect to event. A few algorithms and methods are available to detect events and to discover clusters from social media or user-generated content [13] [14]. Event detection from social network data is used to: 1) detect earthquake and broadcast the alarming situation to all nearby potential users [15]; 2) extract social trends [13] and unusual activities [16] from Twitter data, and Flickr data [17]; 3) to forecast popularity for upcoming events [14]; and 4) detect traffic flow and traffic constraints including accidents, road closed or blocked [7]. In this paper, we propose a technique to take advantage of the growing set of live and historical social datasets to identify events of Interest, POI semantics, and also enhance maps with traffic updates and roads semantics from social data, and statistical analysis from open governmental data.

III. HIGH-LEVEL ARCHITECTURE

We present a smart map building framework that retrieves data from multiple sources, processes that data in order to find knowledge-based layers and visualizes those layers on maps.

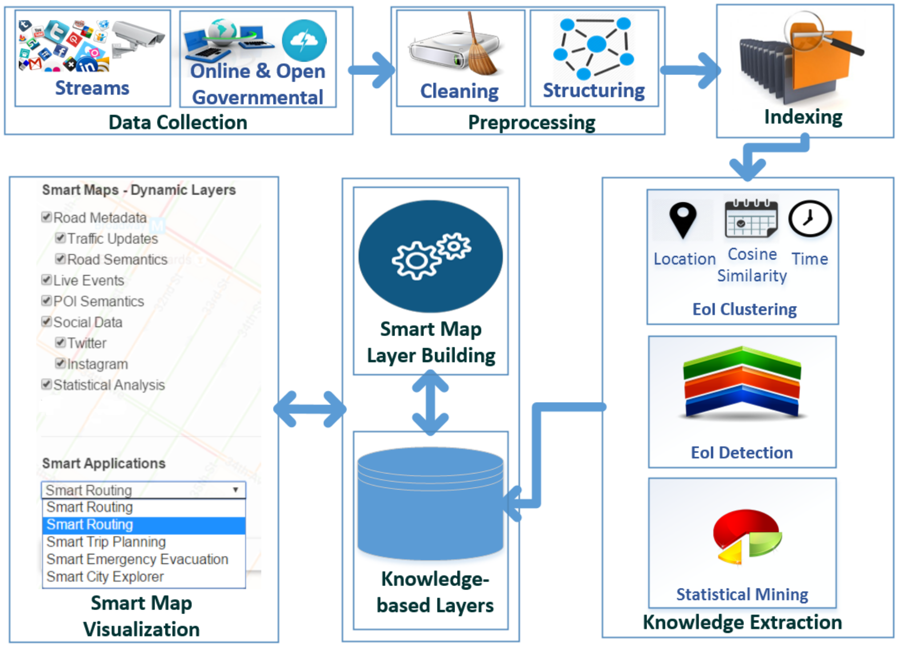

Figure 2. Architecture of the smart maps framework

This framework collects microblog social data streams, open government data, and online data under one platform in order to provide relevant knowledge to users. This helps users in understanding their surroundings, such as live or upcoming events within cities, traffic accidents, road constructions, temporarily closed paths, POI semantics, as well as offers and discounts. Fig. 2 shows an overview of our proposed smart maps architecture with the salient components.

Data Collection and Management Data is retrieved from different data streams, available APIs and online web services. Multiple techniques and policies (frequency, format, structured or unstructured) should be applied on each source of data to be collected. Following are the three ways that we used to collect disparate data from multiple sources:

- Data chunks: In data chunks mode, we download files from different source links that contain partial or full datasets. This data can then be imported or converted into another format for further processing (e.g., crime and accident statistical data). For our use case, we used datasets provided by authorities or open data for different cities, such as New York, USA (including area boundaries, transportation networks, geological data, resources etc.), and other crowdsourced data from Open Street Map and Yelp.

- Single Query: In a single query mode, we used an interface to fetch data by using a single query. This can be achieved by using Restful API to retrieve data in XML and JSON (e.g., Google Traffic and Weather API).

- Continuous Query: In a continuous query, we run crawlers that collect streams of data for each data source (i.e., Twitter and Flickr). Data in this mode is retrieved through Restful APIs of social networks. This mode requires specific handlers for each source of data.

Preprocessing: In the preprocessing step, data cleansing is required by removing noise and irrelevant fields and, in

{"**header**":{"**time**":"Fri Jan 01 00:39:11 +0000 2016","**geo**":{"**type**":"point","**coordinates**":[34.84863834,-95.54509333]}, "**source**": "twitter", "**eventType**":["Party"], "**type**":"text","**potentialEvent**":"true", "**eventname**":"Social events/party","**country**":"United States","**city**":"Oklahoma"}, "**payload**":{"**tags**":[],"**id**":"2605873770","**followers**":"581","**title**":"","**text**":"Party @ Stacey's 😊", "**viewers**":"0", "**screenName**":"TheLedgenBeare", "**language**":"en", "**displayName**":"Ledgen","**url**":null}}

Figure 3. Sample Data Packet from Twitter Streams

case of social data, converting semi- or unstructured data into a structured format referred to as packets. Each packet has a meta-data header, containing source, location, time, type, potential event and event properties, and a payload, containing the actual content (see Fig. 3). Based on the list of predefined event corpus database, useless data will be discarded and potential packet marked as true. For statistical data, we clean them by removing irrelevant fields. For EoI detection, we used NLP techniques for tokenization and to identify Part-of-Speech by taking into account stop words, out of vocabulary words and other abbreviations, such as '*i know you*' (*iky*).

Indexing: Preprocessed data needs then to be stored and indexed for further data integration and clustering. Spatio-temporal indexing schemes for efficient retrieval of queries are implemented. As we are dealing with geo-tagged data for the whole world, we propose a hierarchical data structure (similar to a partial quad tree [18]) that helps in the efficient processing and clustering by comparing packets within leaf cells (i.e., nearby geotagged data packets). This approach divides the world into cells at different levels of granularity based on the number of data points. Geo-tagged streams, Yelp, OSM and statistical data pieces are tagged within a particular *cellID*. We design two types of indexers: 1) a spatial geohash indexer for spatial raw data packets; and 2) an knowledge-based indexer for extracted events and layers data.

 ISBN: 978-1-61208-539-5

	yelpid character varying(100)	poiname character varying(100)	twitteraccount character varying(40)
1	bryant-park-new-york-2	Bryant Park	bryantparknyc
2	red-dawn-combat-club-fresh-meadows-4	Red Dawn Combat Club	RedDawnBJJ
3	queens-botanical-garden-flushing	Queens Botanical Garden	queensbotanicl
4	harlem-yoga-studio-new-york	Harlem Yoga Studio	harlemyoga
5	gleasons-gym-brooklyn	Gleason's Gym	Gleasonsboxing
6	stone-street-tavern-new-york	Stone Street Tavern	stonesttavern
7	the-royal-palms-shuffleboard-club-brooklyn	The Royal Palms Shuffleboard Club	RoyalPalmsClub
8	the-fashion-class-manhattan-2	The Fashion Class	TheFashionClass
9	brother-jimmys-bbq-new-york	Brother Jimmy's BBQ	BrotherJimmys
10	the-bahche-brooklyn-4	The Bahche	The Bahche

Figure 4. PostGIS screenshot to demonstrate data integration of POIs from Yelp ID and Twitter Screen Name

A. Knowledge Extraction

Text mining and clustering techniques are used to find Events of Interest and to infer statistical analysis, such as accident and crime prone places. We extract time, location and identify text similarity to detect the type of EoI. For statistical analysis, we find nearby roads and points of interest that are unsafe (i.e., accident prone and hot crime zones by a certain threshold). We have also divided the knowledge extraction module into the following three levels:

- Direct Detection: In this level, data from trustworthy or authorized sources are considered. We can find authentic social accounts or feeds related to traffic, incidents, etc., in order to classify POIs and to detect social events from trustworthy sources. Most of the announcements related to EoI, such as offers, discounts, upcoming and ongoing events, are published by PoI owners, so it is important to identify authentic sources in social data so that these events can be reported. As illustrated in Fig. 4 and during the preprocessing phase, the system identifies Twitter screen names and Yelp identifiers of trustworthy sources by using location and string matching techniques between social data and POI data collected from Yelp and OSM.

- Indirect Detection and Mining: The second level is to apply mining techniques on fused data from multiple sources in order to extract EoI semantics from unspecified happenings. This level is more complex as compared to the first level, as here we need to consider a truth probability model. Our system adopts the graph analogy where each potential event stream is considered as a *node* and the value of 'cosine similarity using term frequency - inverse document frequency(TF-IDF)' between streams as a weight of the bidirectional *edge*. Data packets with a high text similarity value are clustered using the DBSCAN clustering algorithm. The DBSCAN is suitable in our approach as, unlike most of the other clustering methods, it does not require a prior knowledge of the minimum number of clusters. It will help to detect unspecified events and the hot topic detection as well.

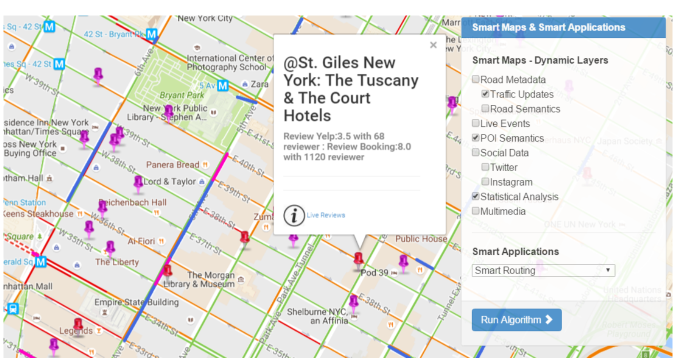

Figure 5. Overview of implemented smart maps

- Statistical Detection and Mining: The third level is performed on historical open governmental data. We apply mining and detection techniques in order to extract statistical analytics, mainly, accident prone areas and crime hotspots. Clustering of crime and accident data is performed within each cell of the hierarchical index tree, and based on a predefined threshold value, edges and POIs inside that area are marked as a crime/accident prone area. We use this approach to find safest path (see Section IV).

B. Knowledge-based Layers

Extracted knowledge will then be spatio-temporally indexed in a knowledge-based layers database. Each EoI is tagged with the cell Identifier, and pieces of statistical analysis are associated with segments or nodes of the road networks. For temporal aspects and cleaning of expired EoIs, we took two parameters: a) *'birth time'* that indicates the first existence of the event in our system, whenever we calculate the first cluster of packets related to that event; and b) *'time of occurrence'* that marks the actual happening time of the event (e.g., next Monday). We clean the EoI from our data after the *'time of occurrence'* has expired using a combination of piggy back and periodic approach. Periodic verification is used to check expired events periodically, whereas piggy back approach is to check for expired events whenever we get any new event within the same cell in the hierarchical tree.

C. Smart Map Layer Building

This component is used to fetch index events from main memory and/or disk. It has two main components; a) query optimizer and b) query engine. The main task of the query optimizer is to find the best query plan. It handles predefined queries related to existing layers displayed on map. The query engine retrieves the plan from query optimizer, so that sub-queries to fetch data from the main memory or disk can be performed. Finally, the smart map layer builder accumulates different results and sends it to the visualizer.

D. Generating and Visualizing Smart Map Layers

The visual interface provides a rich set of spatio-temporal dynamic layers on top of existing maps. The visual renderer interacts with the map building engine to run queries including range, k nearest neighbor (K_{NN}), aggregated and routing

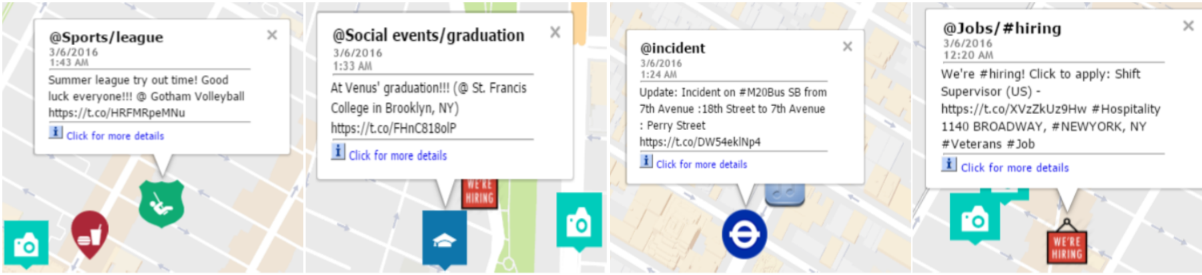

Figure 6. Enhanced city explorer on smart maps: Different type of EoIs on 3rd June: sports, graduation party, road incident, jobs hiring (from left to right)

queries. Fig. 5 Illustrates the visualization of implemented smart maps with POI semantics (displayed as marker), traffic updates and statistical information (blue and pink thick color lines demonstrating accident prone and crime prone roads)

IV. Implementation and Results

To validate our approach, we developed a prototype based on 30Million+ geotagged tweets of world and Yelp, Booking.com, Open Statistical Data from USA governmental website, OSM road network and OSM POI list of New York, USA. The front-end is a web-based application that is used to visualize *smart maps* with dynamic layers and providing a interface to perform the queries related to enhanced routing and city explorer. We are using i7-4712 HQ-CPU @ 2.30GHz with 16GB DDR-2 RAM for in the back-end for processing using the following libraries and software:

- osm2pgsql: The Open Street Map component contains planet dump data that is converted into PostGIS datasets by using the osm2pgsql (github.com/openstreetmap/osm2pgsql) tool.

- *PostGIS*: We installed PostGIS for spatial query over PostgreSQL to store road network data, POIs extracted from Yelp, Booking.com and OSM with their semantics.

- *NLP*: We used Ark-tweet-NLP [19] library for part-of-speech and annotation. This library is trained for Twitter and produce better results than Stanford NLP. It takes care of the out-of-vocabulary words, and stop words used in Twitter.

- *Clustering Algorithm*: Data packets with a high text similarity value are clustered using DBSCAN clustering algorithm. The DBSCAN is suitable in our approach as, unlike most of the other clustering methods, it does not require a prior knowledge of the minimum number of clusters.

- *Taghreed Crawler* [10]: The same crawler was used to collect geotagged tweets.

- *OsmPoisPbf*: We used OsmPoisPbf (github.com/MorbZ/OsmPoisPbf) tool to extract POIs from OSM in PBF file format.

- *Other libraries*: Apart from above, we used Jackson JSON and Yelp, and the Brezometer weather REST APIs in our system. OSMPgRouting A* algorithm used to calcuate the path and we updated edge values using google maps API in case of traffic and increase the weight of edges based on crime and accident frequencies. Back-end implementation is done in Java 1.7.

Following are the "City Explorer" and "Routing" applications as demonstrated on top of the smart maps framework:

TABLE I. RELATION BETWEEN KEYWORD OF USER'S QUERY, EOIS AND SOURCES OF DATA COLLECTION

Keywords	Layers	Sources
Safe	Accident (Prone Edges)	Open Government Data
Safe	Crime Hotspots	Open Government Data
Music Concert	Social Events	Social Data and News
Discounts	POI Announcements	Social Data

A. City Explorer

Fig. 6 illustrates an example output by showing different type of EoIs a) 'Sports League'; b) 'Graduation Party; c) 'Traffic Incident'; and d) 'Jobs Opening'. This allows us to show live events that provide latest information collected from multiple sources about historical places, touristic places, dining, events, shops, news, live tweets, weather updates, traffic updates, semantics of points of interest, and visualize multimedia information that enhance the system usability.

B. Routing

For a given spatio-temporal area, we use the *smart maps* system to enhance traditional queries. We grouped extracted EOIs including social gatherings, 'hotspots' and 'hot times' in a criminal activity, POI semantics, accident-prone roads, in order to provide enriched response of users' routing queries. For example, Consider a city dweller who is interested in current ongoing activities in her neighborhood district and she posts the following query , *"Show safe and fastest path to find the nearest music concert with some ticket discounts"*. Currently, existing maps are unable to provide answers to this type of queries. *Smart maps* framework can support such queries, by determining the relationship between the list of keywords in that query, and the corresponding layers and sources of data (cf., Table 1). Fig. 7 shows the results of the query: a) a map with crime (pink color) and accident prone (blue color) edges; b) an optimized path with respect to time between two markers (showing with red color), c) a crime free path by avoiding crime prone edges (light green color), whereas the safest path can be calculated by avoiding both accident prone and crime prone edges (showing with blue color).

V. Conclusion

This paper presents a *smart maps* framework, that adds dynamic layers to traditional maps by handling social data

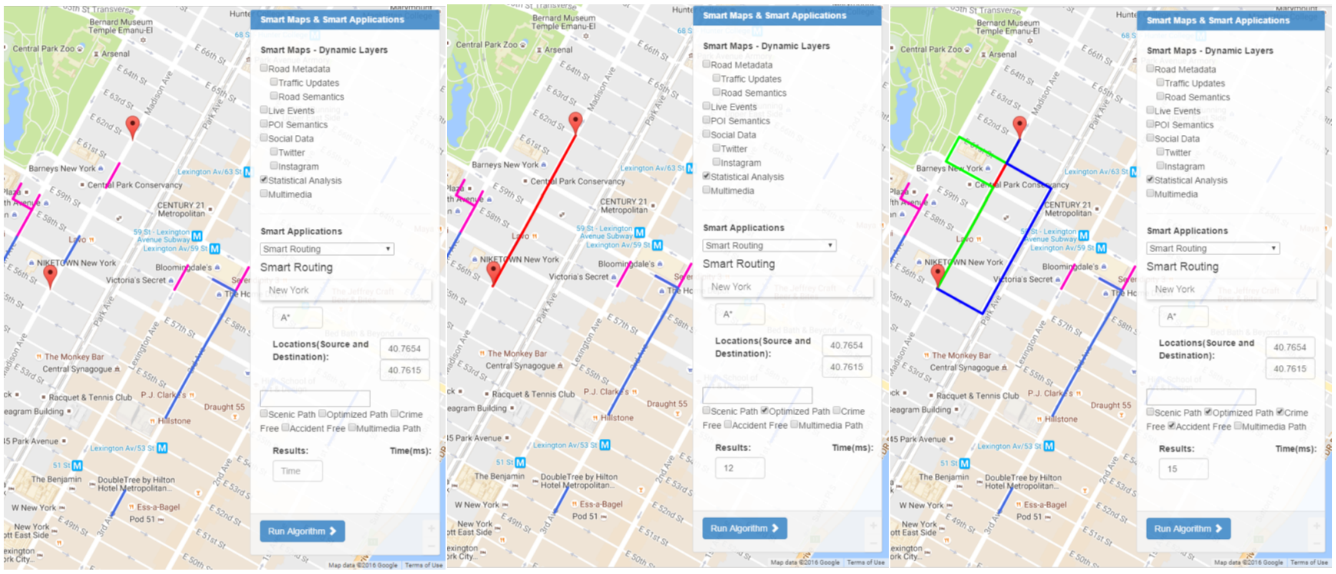

Figure 7. Enhanced routing on smart maps with different options, such as, optimized, crime free, accident free.

streams, and by developing different algorithms for the efficient extraction, clustering, and mapping of live crowd-sourced events. This framework wraps incoming unstructured data streams into data packets, that is, a generic structured format of a potential event. These packets are then processed to extract EoIs based on different dynamic layers. This framework helps in enhancing existing spatial queries including city explorer and routing using extracted knowledge of the dynamic layers. This platform can be easily enriched with new data sources, such as online newspapers. The system can provide valuable knowledge to authorities, governments, market firms, POI owners, event organizers, and end-users in decision making, thus enhancing infrastructure and human life style. Our approach is scalable but not tested for the whole world. In future, we are planning to work on a big data platform to validate scalability and performance factors. Furthermore, we plan to add more data sources to increase completeness, correctness, and conciseness of detected events.

References

[1] A. Halevy, A. Rajaraman, and J. Ordille, “Data integration: The teenage years,” in Proceedings of the 32Nd International Conference on Very Large Data Bases, ser. VLDB ’06. VLDB Endowment, 2006, pp. 9–16.

[2] S. Consoli, M. Mongiovic, A. G. Nuzzolese, S. Peroni, V. Presutti, D. Reforgiato Recupero, and D. Spampinato, “A smart city data model based on semantics best practice and principles,” in Proceedings of the 24th International Conference on World Wide Web, ser. WWW ’15 Companion. New York, NY, USA: ACM, 2015, pp. 1395–1400.

[3] “Smarter planetsmarter cities. http://www.ibm.com/smarterplanet/us/en/smarter_cities/overview,” 2016.

[4] “Citynext. https://partner.microsoft.com/en-us/solutions/citynext,” 2017.

[5] X. L. Dong and F. Naumann, “Data fusion: Resolving data conflicts for integration,” Proc. VLDB Endow., vol. 2, no. 2, Aug. 2009, pp. 1654–1655.

[6] G. Lamprianidis, D. Skoutas, G. Papatheodorou, and D. Pfoser, “Extraction, integration and exploration of crowdsourced geospatial content from multiple web sources,” in Proceedings of the 22Nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, ser. SIGSPATIAL ’14. New York, NY, USA: ACM, 2014, pp. 553–556.

[7] F. U. Rehman et al., “Toward dynamic path recommender system based on social network data,” in Proceedings of the 7th ACM SIGSPATIAL International Workshop on Computational Transportation Science, ser. IWCTS ’14. New York, NY, USA: ACM, 2014, pp. 64–69.

[8] M. F. Goodchild, “Citizens as sensors: the world of volunteered geography,” GeoJournal, vol. 69, no. 4, 2007, pp. 211–221.

[9] J. Krygier and D. Wood, Making maps: a visual guide to map design for GIS. Guilford Press, 2011.

[10] A. Magdy et al., “Taghreed: a system for querying, analyzing, and visualizing geotagged microblogs,” in Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, 2014, pp. 163–172.

[11] H. Samet et al., “Reading news with maps by exploiting spatial synonyms,” Commun. ACM, vol. 57, no. 10, Sep. 2014, pp. 64–77.

[12] “Statista: The statistics portal. http://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/,” 2016.

[13] S. B. Kaleel and A. Abhari, “Cluster-discovery of twitter messages for event detection and trending,” J. Comput. Science, vol. 6, 2015, pp. 47–57.

[14] X. Zhang, X. Chen, Y. Chen, S. Wang, Z. Li, and J. Xia, “Event detection and popularity prediction in microblogging,” Neurocomputing, vol. 149, 2015, pp. 1469–1480.

[15] M. Avvenuti, S. Cresci, A. Marchetti, C. Meletti, and M. Tesconi, “Ears (earthquake alert and report system): A real time decision support system for earthquake crisis management,” in Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD ’14. New York, NY, USA: ACM, 2014, pp. 1749–1758.

[16] R. Lee, S. Wakamiya, and K. Sumiya, “Discovery of unusual regional social activities using geo-tagged microblogs,” World Wide Web, vol. 14, no. 4, Jul. 2011, pp. 321–349.

[17] M. Schinas, S. Papadopoulos, G. Petkos, Y. Kompatsiaris, and P. A. Mitkas, “Multimodal graph-based event detection and summarization in social media streams,” in Proceedings of the 23rd ACM International Conference on Multimedia, ser. MM ’15. New York, NY, USA: ACM, 2015, pp. 189–192.

[18] H. Samet, “The quadtree and related hierarchical data structures,” ACM Computing Surveys (CSUR), vol. 16, no. 2, 1984, pp. 187–260.

[19] K. Gimpel et al., “Part-of-speech tagging for twitter: Annotation, features, and experiments,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2, ser. HLT ’11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 42–47.

 ISBN: 978-1-61208-539-5