FRAGOLA: Fabulous RAnking of GastrOnomy LocAtions

Ana Alvarado, Oriana Baldizán, Marlene Goncalves, and María-Esther Vidal

Universidad Simón Bolívar, Venezuela
{aalvarado,obaldizan,mgoncalves,mvidal}@ldc.usb.ve

Abstract. Nowadays, large open datasets are frequently accessed to select, for example, restaurants that best meet gastronomy criteria and are closer to their current geo-spatial locations. We have developed a skyline-based ranking approach named FOPA, which is able to efficiently rank resources that fullfil this type of multi-objective queries. As a proof of concept, we developed FRAGOLA (Fabulous RAnking of GastrOnomy LocAtions), a tool that implements FOPA and ranks gastronomy locations based on multi-objective criteria. We will demonstrate FRAGOLA, and attendees will observe scenarios where FOPA overcomes performance of existing skyline-based approaches by up to two orders of magnitude.

1 Introduction

Under the umbrella of the Semantic Web and the Open Data initiatives, large datasets have been published and can be publicly accessed from any node of the Internet. Although the democratization of the information provides the basis to manage large volumes of data, there are still applications where it is important to efficiently identify only the best tuples that satisfy a user requirement. Particularly, large datasets of government and private recreational data are available, and these data can be accessed to identify the places that best meet users' cuisine requirements and are closer to her current geo-spatial location. Based on related work, we devised a solution to this ranking problem and developed techniques able to identify the gastronomy locations that best meet these multi-objective queries, i.e., the gastronomy locations are not better than other gastronomy locations in terms of the multi-objective criteria. The set of non-dominated points is known as *skyline*, i.e., set of points such that, none of them is better than the rest [2,3]. We developed an algorithm that combines ideas from the approaches described by Balke et al. [2] and Chen et al. [3] to compute the skyline points that best meet a multi-objective query; the algorithm implements different pruning criteria and avoids traversing the whole space of data to compute the skyline. Thus, the execution time as well as the number of probes required to output the answer is minimized. We illustrate the performance of our approach on a dataset of gastronomy locations downloaded from Zagat,[1] where restaurants are

[1] http://www.zagat.com/paris

Y.T. Demey and H. Panetto (Eds.): OTM 2013 Workshops, LNCS 8186, pp. 408–413, 2013.

characterized by six parameters, and queries required to rank the best restaurants expressed in terms of these parameters as well as with respect to different geo-spatial locations. The demo is published at `http://fragola.ldc.usb.ve/`.

2 The FRAGOLA System

An RDF document is comprised of triples that describe resources in terms of several properties; formally, this can be seen as a set of multi-dimensional points that describe each resource in terms of their properties. A multi-objective *query* is comprised of: *i*) a *condition* or list of RDF properties, and the **MIN** or **MAX** directives indicating if the values of the corresponding property must be minimized or maximized, and *ii*) the user current location. The *answer of a query q* corresponds to the points or resources in the multi-dimensional dataset D that are incomparable, i.e., the *skyline* that is composed of all the points p, such that: *i*) there is not other point p' in D with values better or equal than p in all the attributes of p, and *ii*) other points in the skyline are better than p in at least one attribute. The problem of computing the skyline is polynomial on the size of the dataset, and the goal of state-of-the-art skyline algorithms is to compute the set of incomparable points without having to perform a polynomial number of comparisons [4,5]. The FRAGOLA System seeks to illustrate how our Final Object Prunning Algorithm (FOPA) [1] achieves this goal by using some data properties, and extending features of the algorithms RSJFH [3] and IDSA [2].

These three algorithms assume that the data is stored following a vertically partitioned table representation, i.e., for each dimension or RDF property a, there exists a relation aR composed of two attributes, *Subject* and *Value*; tuples are ordered according to the attribute *Value*. Further, indices are kept on top of these tables to provide direct and sequential access, and a data structure is used to track the last values of the objects seen in each dimension. The algorithms work on iterations, where the best entry(ries) in each of the vertical tables is(are) considered in one iteration. The goal of the three algorithms is to minimize the number of comparisons between data dimensions to compute the skyline. First, the RDFSkyJoinWithFullHeader (RSJFH) algorithm proposed by Chen et al. [3], uses the data structure named header point, to record the worst values of the tuples explored in previous iterations; this information is used to guide the pruning of tuples seen in future iterations. Second, the Improved Distributed Skyline Algorithm (IDSA) proposed by Balke et al. [2] guides the search into the space of the final object, i.e., an object that has been considered in all the vertical tables; once the final object is found IDSA can ensure that a super-set of the skyline has been found, and a post-processing step is fired to discard the tuples in the super-set that are not incomparable. Although experimental studies reported in the literature [2,3] suggest that RSJFH and IDSA are efficient, both may suffer of the following drawbacks:

i) RSJFH produces incomplete results for multi-objective criteria of three or more dimensions, and *ii*) IDSA performs poorly when the final object is comprised of the worst value of at least one dimension. To overcome these limitations,

we propose the skyline algorithm FOPA that assumes the following: *i*) tuples in a dataset are characterized by multi-dimensions and *ii*) tuples are stored following the vertical partition approach ordered and indexed by two indices.

Additionally, FOPA maintains information about the last values seen so far, and uses this information to guide the search of the final object. The algorithm also uses a correct pruning strategy in order to discard within the group of objects read in a given iteration, the ones that will not be part of the skyline. This allows FOPA to compute the skyline set as soon as it finds the final object or a dimension has been completely scanned, not incurring in any further comparisons. Thus, FOPA can ensure completeness while the number of comparisons and data accesses is reduced. As a proof of concept, we implemented FOPA, RSJFH and IDSA in the FRAGOLA system on top of the multi-dimensional dataset of restaurants in Paris that is provided by Zagat. Our goal is to illustrate the performance and behavior of these three algorithms.

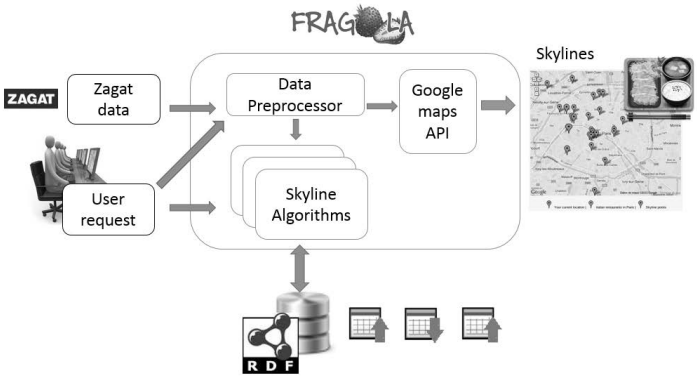

Fig. 1. The FRAGOLA Architecture

The *restaurant ontology*[2] is used to describe each restaurant, while Geonames[3] is used to describe restaurants' geo-spatial locations. FRAGOLA receives multi-objective queries that express conditions over characteristics (RDF properties) of the restaurants and users' current geo-spatial locations. The answer to a multi-objective query is the set of incomparable restaurants that comprises the skyline with respect to the attributes and directives considered in the query. FRAGOLA is comprised of the following components: a data preprocessor, the skyline engine, and Google maps' API.

The data preprocessor transforms the data provided by Zagat into a [0,1] scale and calculates the distance between the users' current geo-spatial location and that of each restaurant. The user's request allows the selection of the dimensions (RDF properties) of interest for the ranking algorithms. The processed data is then stored in vertically partitioned tables and the triples are ordered in terms of

[2] `http://schema.org/Restaurant`
[3] `http://www.geonames.org/`

the property values. A skyline engine implements the three skyline algorithms: IDSA, developed by Balke et al. [2], RSJFH, developed by Chen et al. [3], and FOPA. The Google map API is then used to visualize the resulting restaurant skyline set in the map. Additionally, FRAGOLA reports on the algorithms' performance in terms of the number of readings, vertically partitioned table joins, comparisons, prunnings, and the size of the skyline.

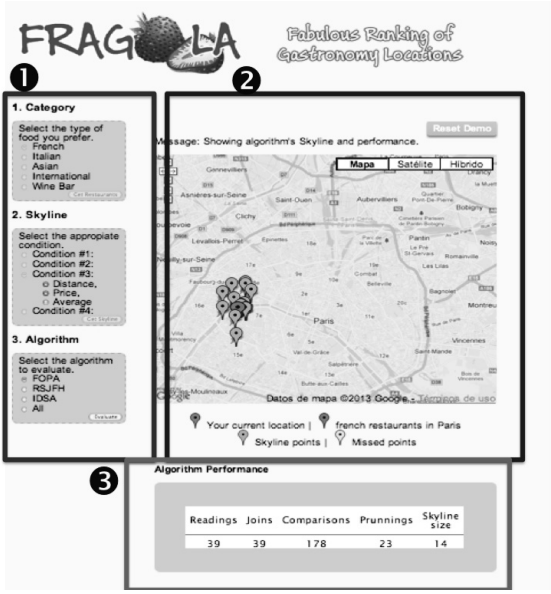

Fig. 2. The FRAGOLA GUI for the ZAGAT dataset. 1-Query Area (Three Steps: 1. Category; 2. Skyline; 3. Algorithm. These steps need to be executed in order); 2-Google Map that reports the Skyline Restaurants; 3-Metrics on the Algorithms' Performance.

Figure 2 shows the FRAGOLA interface for the ZAGAT dataset of restaurants in Paris. The area enclosed inside the rectangle number 1 shows the query interface; where the user can select restaurants of different categories, select four multi-objective queries, and compare three algorithms. Since FRAGOLA is a demo and the data used has been previously calculated the user must make each of his selections effective by clicking the "Get Restaurants", "Get Skyline" and "Evaluate" buttons after each selection, in this order. Results are reported in the Google map window in the area enclosed by the rectangle number 2; where the user's current location (fixed in Paris) is highlighted in blue, the skyline restaurants in green, and the restaurants in the skyline that cannot be found by the RSJFH algorithm are reported in yellow as missed skyline points. The user can reset the demo at all times using the "Reset Demo" button enclosed in the blue rectangle. Finally, performance is reported in the area enclosed by the rectangle 3.

3 Demonstration of Use Cases

As of February 2013, ZAGAT for Paris contains 481 restaurants, described in terms of: address, geo-spacial location[4], food category and quality, decoration, service quality, average price, overall average, and popularity index; the corresponding RDF document is comprised of 7,696 triples. We will show the impact of the dataset and skyline size on the performance and completeness of the three algorithms. First, attendees will be able to select different categories of restaurants, then, different multi-objective queries, and finally, the different algorithms. We will illustrate the location of the skyline restaurants with respect to the user current location, and the rest of the properties of these restaurants; additionally, the attendees could be able to observe values of the different metrics that measure the performance and behavior of the selected algorithms.

We will show that for the non-selective restaurant food category criterium, i.e., the french, the multi-objective query represented by *Condition 1*[5], FOPA produces the complete skyline, i.e., 13 restaurants, while the number of readings and joins is reduced by one order of magnitude and the number of comparisons is decreased by almost three orders of magnitude. RSJFH overcomes IDSA, but is only able to produce the complete answer when the skyline is small, e.g., skyline of 2 points in *Condition 2*. Otherwise, RSJFH produces incomplete answers and never overcomes the performance of FOPA. This poor performance with respect to FOPA is because RSJFH always needs to traverse at least one of the partitioned tables completely, and the number of readings, joins and comparisons increase with the size of the database. On the contrary, FOPA resembles IDSA and stops when a resource is seen in all the partitioned tables; additionally, it uses information about the resources seen in previous iterations to avoid accessing the same resource multiple times. Thus FOPA performs less number of comparisons and is able to build the skyline before traversing one of the tables completely; outperforming both RSJFH and IDSA while it generates the whole skyline.

4 Conclusions

We have stated the problem of identifying the nearest locations that best meet a set of characteristics as a skyline-based ranking problem, and proposed an algorithm that provides an efficient solution to this problem. As a proof of concept, we have developed FRAGOLA and implemented the proposed skyline algorithm on top of a dataset of restaurants annotated with geo-spacial information. We demonstrate the capabilities of the ranking techniques as well as the performance of our solution with respect to state-of-the-art solutions. Results suggest that FOPA overcomes other approaches by up to two orders of magnitude.

[4] Geo-spacial annotations were done manually.

[5] Condition 1 represents the restaurants that are incomparable with respect to minimal distance to the current location, minimal price and maximal popularity.

References

1. Alvarado, A., Baldizan, O., Goncalves, M., Vidal, M.-E.: Fopa: A final object pruning algorithm to efficiently produce skyline points. In: Accepted at DEXA (2013)
2. Balke, W.-T., Güntzer, U., Zheng, J.X.: Efficient distributed skylining for web information systems. In: Bertino, E., Christodoulakis, S., Plexousakis, D., Christophides, V., Koubarakis, M., Böhm, K. (eds.) EDBT 2004. LNCS, vol. 2992, pp. 256–273. Springer, Heidelberg (2004)
3. Chen, L., Gao, S., Anyanwu, K.: Efficiently Evaluating Skyline Queries on RDF Databases. In: Antoniou, G., Grobelnik, M., Simperl, E., Parsia, B., Plexousakis, D., De Leenheer, P., Pan, J. (eds.) ESWC 2011, Part II. LNCS, vol. 6644, pp. 123–138. Springer, Heidelberg (2011)
4. Fuhry, D., Jin, R., Zhang, D.: Efficient skyline computation in metric space. In: EDBT, pp. 1042–1051 (2009)
5. Skopal, T., Lokoc, J.: Answering metric skyline queries by pm-tree. In: DATESO, pp. 22–37 (2010)