Recent Progress Towards an Ecosystem of Structured Data on the Web

Nitin Gupta, Alon Y. Halevy, Boulos Harb, Heidi Lam, Hongrae Lee, Jayant Madhavan, Fei Wu, Cong Yu

Google Research
U.S.A.

***Abstract*—Google Fusion Tables aims to support an ecosystem of structured data on the Web by providing a tool for managing and visualizing data on the one hand, and for searching and exploring for data on the other. This paper describes a few recent developments in our efforts to further the ecosystem.**

I. Introduction

The combination of a rich repository of structured data on the Web coupled with new tools for data management and visualization are leading us to exciting times in which structured data is having a profound impact on many aspects of our lives. In many countries, citizens take for granted the fact that governments, local authorities, and non-government organizations should make a variety of data sets available to the public. These data sets span a variety of topics such as economic indicators, crime statistics, educational data, government spending and campaign contributions. The new generation of tools for managing and visualizing data have empowered data activists, led by journalists, who are turning this data into visualizations and stories that are spread by social networks and seen by millions of people [1]. These visualizations, stories and public attention, in turn, lead to new questions and hence a demand for additional data.

The success of this trend is still dependent on improving our solutions to several long-standing data management problems. First, we need to continue developing tools that enable a broader set of users to manage data and create compelling visualizations. Second, we need methods for identifying high-quality data from the Web and other corpora. Third, we should be able to recover the semantics of these data sets sufficiently well so they can be displayed for relevant user queries and combined with other data sets to provide additional meaningful insights. All put together, we need to create an ecosystem of tools and data that enable us to discover good data, create useful artifacts from it, and contribute it back to the Web.

The Google Fusion Tables project has been addressing some of these issues. At the core, we offer a cloud-based tool for querying, sharing, visualizing, integrating and publishing data. We complement the tool with a search engine that enables users to find high-quality tables from a corpus of over 130 million tables on the Web. This paper highlights some of the new functionalities we recently added to our service.

II. Data management

Google Fusion Tables is a cloud-based service for data management and visualization. With Fusion Tables, it is easy to upload data, share the data with collaborators or make it public, and to pose simple queries. Fusion Tables also emphasizes the ability to create visualizations of the data that can be easily embedded in other Web sites. Because of its ease of use, it has been adopted by *data enthusiasts*, namely individuals and organizations that have valuable data they want to share or visualize but do not have deep technical expertise. For example, journalists use Fusion Tables very frequently to include data in their articles. In addition, Fusion Tables has been used in disaster response situations where valuable data has been made available to people in an area of need; to cover the election results in several countries around the world and for novel crowd-sourcing applications. A few examples of Fusion Tables applications is illustrated in [2].

Initially, Fusion Tables invested heavily in creating map visualizations. The reason for this investment was to respond to specific requests of our users and because maps are the most common visualization and applies to a wide range of domains. However, geographical information systems are an entire field onto themselves, and rather than investing wholly in maps, we decided to diversify into other visualizations. Hence, in the recent months we have launched visualizations such as the zoomable time-line (see Figure 1) and the network graph (see Figure 2).

Importantly, the architecture and optimizations we created for map visualizations (e.g., [3]) informed us on how to support other visualizations. The main challenge we had to address is providing an efficient and smooth visualization experience in a cloud-based environment, where users expect immediate responses as they zoom into a visualization or pan across it. The specific challenge is that while the underlying data sets may be large, only a small amount of data can be transmitted to the client to satisfy the performance requirements and not overload the client. Hence, we built a hierarchical index that guarantees that with every operation we transmit only a bounded number of rows and these can be determined very efficiently from the index.

In many cases, our users were not familiar with the data sets they were exploring. Instead of looking for a specific fact, they might be looking for patterns in the data, either ones that apply

978-1-4673-4910-9/13/$31.00 © 2013 IEEE

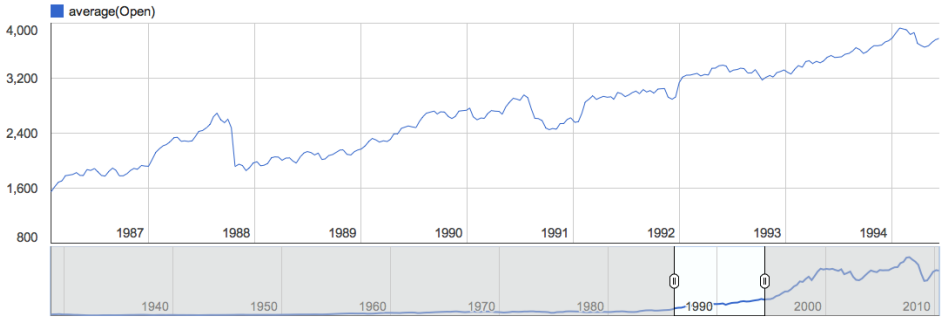

Fig. 1. A zoomable line graph. Users can zoom and pan the line graph in a similar fashion to exploring a map. The line graph shows the daily closing price of the Dow Jones Index for a few decades in the 20th century.

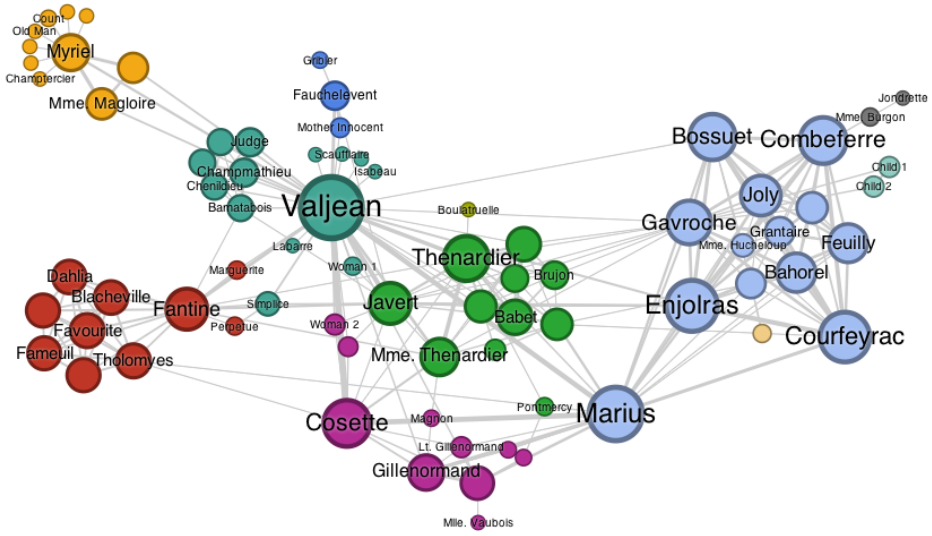

Fig. 2. A network graph displaying relationships between nodes. This graph shows connections between characters in the Les Miserables novel.

to the dataset or idiosyncrasies in the data. Further, they are most likely not aware apriori of the the pattern that they might be looking for.

In Fusion Tables we offer faceted browsing as one possible means of supporting data exploration. Users can select columns to filter their data. For each column, we provide a list of distinct values and counts. When a user selects one (or more) of the values, only the matching rows are displayed. Most interesting, the facets update with every user action, i.e., selecting a value of a column, will result in the facets for the other columns (the values and the counts) being updated to reflect the new condition.

Maintaining accurate facet counts with every user action could be thought of as traversing a data-cube for that dataset. When a user selects a new column to facet on, we have to compute counts for an additional column. When a user selects or deselects a column value, we have to re-compute counts for all facet column but with an updated filtering condition. We have the further requirement that the user experience should be interactive. Thus, we have to compute aggregates over combinations of columns over tables with potentially 100s of thousands of rows in a matter of milliseconds. Note that computing the data-cube is a well-studied challenging problem and pre-computation is not possible when there are many columns or when there are updates (both of which we support).

III. Data Integration

As the name of the service suggests, one of the initial focal points of Fusion Tables is to enable users to combine data from multiple sources, helping them unleash the power of the vast sources of data on the Web. Now that some of the more basic aspects of Fusion Tables are relatively stable, we have put more of our effort into data integration.

Keeping in line with our desire to provide an easy-to-use interface, we wanted the experience of data integration, at least in the relatively simple cases, to be much like Web search. In the recently launched merge interface, the user can begin an integration task when viewing one of her tables. The user can specify a keyword query describing the name of the column she would like to add to her current table (via a join operation). Fusion Tables uses the current table and the keyword query

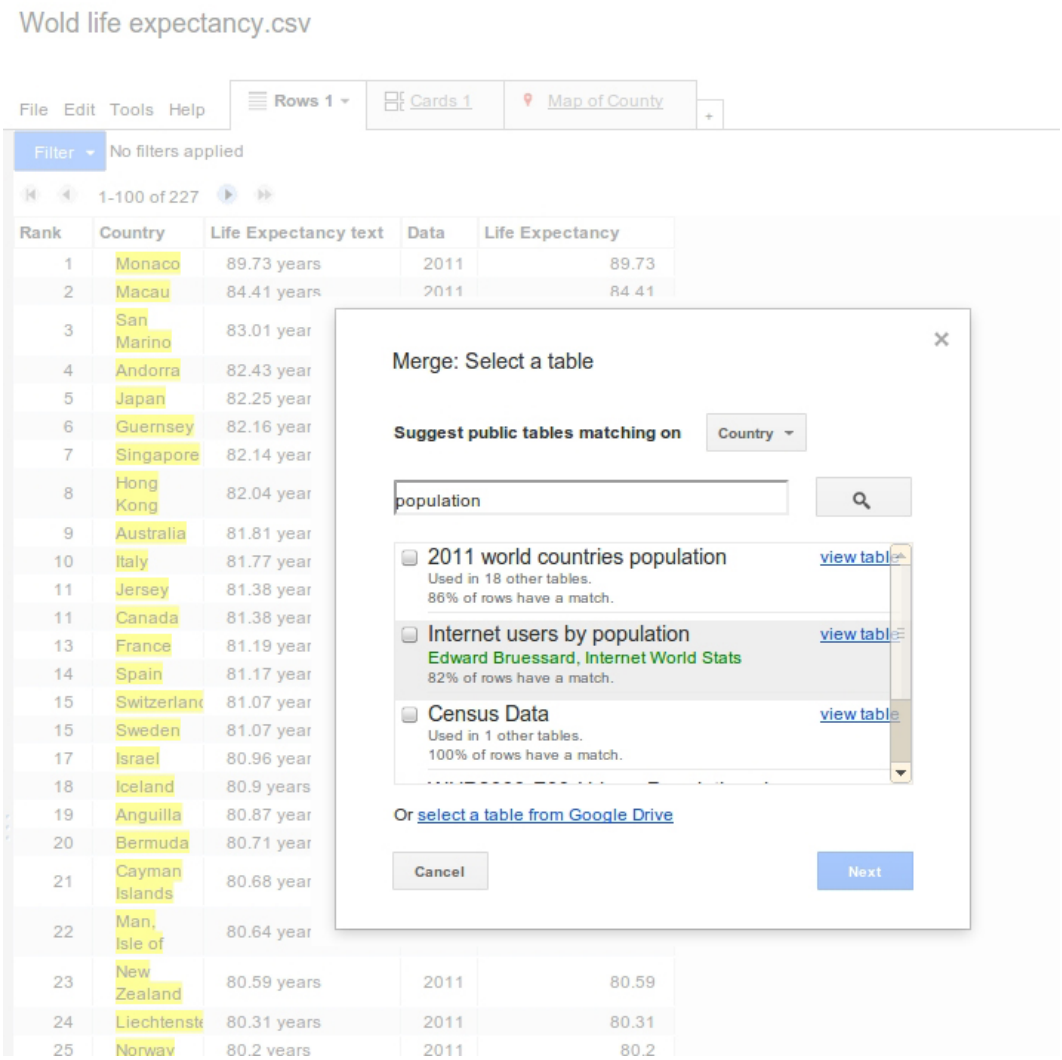

Fig. 3. Merge in Fusion Tables. When looking at a table, the user can pose a keyword query asking for another column. The merge service will find tables that have that column and can join with the current table.

to find other relevant tables. These tables may be drawn from Fusion Tables that were made public or from our corpus of HTML tables on the Web (see Section IV). The retrieved tables will be ranked depending on (1) their relevance to the requested attribute and (2) whether the found table can join with the current table. For example, if the user is looking at a table with the GDPs of European countries and asking for the property POPULATION, there is no point returning a table that has only the populations of Asian countries.

Using our new merge service, simple merges can be done easily by search without the user even knowing that they are doing a join. After launching this service the number of merged tables tripled almost immediately.

IV. STRUCTURED DATA SEARCH

One of the benefits of cloud-based data management systems is that they facilitate and encourage publication of data sets of broad interest. For example, several governments are already using Fusion Tables to publish a variety of data sets. However, to leverage these new assets, we need to provide powerful search mechanisms that enable users to discover useful data. In the previous section we described the merge feature that lets users discover joinable data in the context of the table with which they are working. In the last year we launched a broader search service that draws upon a corpus of over 130 million HTML tables on the Web and the fusion tables that users have declared as public. We are currently expanding the corpus to include CSV files, spreadsheets and other tabular data on Web pages that are not within the HTML table tags.

Developing such a search service requires addressing several challenges. First, we need to filter the HTML tables on the Web to only those that contain high-quality data (which consist of less than 1% of the tables). We filter the tables using a carefully tuned classifier that is trained with example tables. Second, when we find good tables, we try to recover their

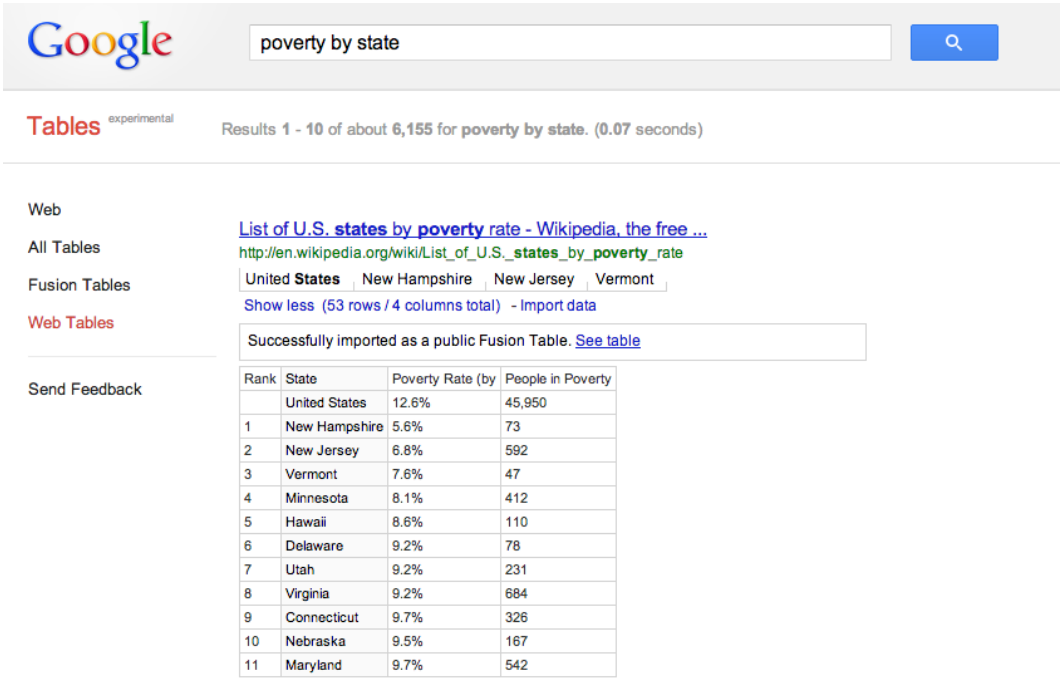

Fig. 4. Search in Fusion Tables. Users can pose a keyword-based data seeking query and get a preview of the table. If the result is a public Fusion Table they can then see the entire table, and if it a table on the Web, then they can examine the table on its original page or import it into Fusion Tables.

semantics: what class of entities are the tables modeling and what properties of these tables are stored in the table. We use the techniques initially described in [4] to recover table semantics. In doing so we leverage Google's Knowledge Graph (which is a superset of Freebase), a repository of high-precision entities and their properties. Following the strategy outlined in [4] we try to find which types in the Knowledge Graph best represents the set of entities in a table.

Merely finding a table using our search service is not enough. Users may want to operate on the found tables and perhaps combine tables in various ways. To support such task flows, we let users import a table found on the Web into Fusion Tables for further processing.

Google's Knowledge Graph contains an enormous amount of high-quality data that is valuable to users of Fusion Tables. We recently *imported* data from Freebase (which is the public portion of the Knowledge Graph) into our table repository so it can be found in our search. To do so, we had to address the challenge of finding appropriate sized and scoped tables to import. More specifically, the data model of the Knowledge Graph is (not surprisingly) a graph. Nodes in the graph represent objects in the world, and their properties are attached to them through edges or paths along edges. Hence, two questions arise. First, which set of entities should go into a single table. For example, putting the set of all movies into a single table may not be as effective as separating movies into multiple tables organized by genre or some other important facet, or even creating a table for that contain only the movies of a particular (popular) actor/actress. The second question is which attributes should be part of a given table. For example, while we may create a table of countries, we certainly do not want a single table with all the attributes of countries. We developed a rule-based mechanism that tries to take into consideration which tables would actually answer user queries.

V. Conclusions

Our work to date has demonstrated that if you provide good tools for data management and visualization, there are plenty of data enthusiasts who are eager to use them to create more value from data and put it in the eyes of the public. However, our work has also uncovered many challenging issues. In particular, understanding the semantics of the data we find is an important challenge as well as identifying the authoritative data on a particular topic. We execpt that going foward we will be able to leverage signals from users to identify useful data and how it can be combined with other data sets.

References

[1] A. Y. Halevy and S. McGregor, "Data management for journalism," *IEEE Data Eng. Bull.*, vol. 35, no. 3, pp. 7–15, 2012.

[2] "Fusion tables gallery," https://sites.google.com/site/fusiontablestalks/stories, 2012.

[3] A. D. Sarma, H. Lee, H. Gonzalez, J. Madhavan, and A. Y. Halevy, "Efficient spatial sampling of large geographical tables," in *SIGMOD Conference*, 2012, pp. 193–204.

[4] P. Venetis, A. Y. Halevy, J. Madhavan, M. Pasca, W. Shen, F. Wu, G. Miao, and C. Wu, "Recovering semantics of tables on the web," *PVLDB*, vol. 4, no. 9, pp. 528–538, 2011.