# Search Beyond the Web: Data from Social Networks and Native Apps

[Keynote Abstract]

Maria Grineva
Yandex Labs
299 South California Avenue
Palo Alto, CA 94306
mariagrineva@yandex-team.ru

## Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; I.2 [**Artificial Intelligence**]: Natural Language Processing—*Language parsing and understanding*

## General Terms

Design, Algorithms

## Keywords

search, social media, social networks, mobile apps

## 1. DATA FROM NATIVE APPS AND SOCIAL NETWORKS

Just a few years ago at the height of the Web 2.0 era, it was hard to believe that a significant amount of user generated data would be created and consumed without the use of the Web. In June 2011, researchers reported that time spent on native apps began to outpace time spent on the desktop or mobile Web [9]. People work, play, communicate and create content using narrowly focused apps, generating data the size of which is now comparable to the size of the Web. For example, Twitter users post 400 million tweets per day, most of them come from mobile clients [5], Foursquare boasts millions of check-ins everyday, more than 5 million photos per day are uploaded to Instagram [3], Spotify users listened to 13 billion songs during the first year the music streaming service was available in the US [1].

Surfing websites through hyperlinks to discover new information or using Web search engine for exploratory queries now seems so ineffective. Instead, we use social networks and specific apps as our daily news source, as well as sources of recommendations for places, music, films and other aspects of our life. This has become possible due to the smart instrumentation of modern social networks: users can connect not only to their friends, but also subscribe or follow interesting people, without being friends in real life, as well as brands, celebrities and news channels.

Typically, data generated in apps are spread via friend connections in social networks. Many apps share their data (explicitly or implicitly) to Facebook. With Facebook Open Graph API [4], an app defines an action that the users perform using the app, as a verb. Then it can publish into the user's timeline, on behalf of the user, thus connecting the current user with an appropriate object instance via the verb. For example in Instagram, a user *takes* a *photo*. In Spotify, a user *listens* to a *song*. Then, every time the user is listening to a song, Spotify published this fact on Facebook on behalf of the user: *"<username> listens to <song name> on Spotify"*. Such frictionless sharing helps apps to reach wider audience virally. It is therefore not surprising that this practice has quickly become very popular: now around 5000 apps share up to 1 billion pieces of content every day via Open Graph API [2].

So far, Facebook doesn't do much with these data: these are too large and noisy to be pushed to users' friends newsfeed. Imagine a stream of all "listens" from your friends on Spotify, Rdio, Pandora, Last.fm and other music apps. So Facebook only shows them in a real-time stream in Ticker for a few seconds. There is no way to access these data just a few days later. While there are obviously a lot of use cases when the user would benefit from being able to access it. Sometimes you want to know what music a particular friend of yours is listening, who of your friends prefers dubstep, or what bars do your friends visit in San Francisco. That is, we need to be able to search over these data.

Speaking about other social networks, only Twitter provides search over tweets inside the user's friends circle, as well as a global search. Still there is no way to search through all the data coming from the user's connections in social networks, in a uniform way.

Collecting data for each user from their different social networks is more complicated than crawling the Web because of the differences in APIs and platforms' policies. However, major social networks support a common practice: the user can give access to their account for an application, and the application can use data from the user's visibility scope and add its value upon these data.

In my talk I will discuss the specifics of the data from social networks and native apps. I will identify challenges in building search for these data and opportunities to build a new and better search experience.

## 2. CHALLENGES IN BUILDING SEARCH

### 2.1 Representing real-world objects in search results

*WIDM'12,* November 2, 2012, Maui, Hawaii, USA.
ACM 978-1-4503-1720-7/12/11.

Native apps typically designed for a specific domain, operate with a particular kind of objects. These objects often represent some real-world entity. For example, users check in at *places* on Foursquare, spot *dishes* with Foodspotting, favorite *tracks* on SoundCloud. Data representing an object is often media-rich: places have its location on the map, photos, tips left by the users who checked in there, if it is a restaurant it can contain information about open hours and menu.

This implies different search result page (SERP) presentation. SERP is no longer a list of ten blue links: users don't have to go to the website to find the sought information themselves. Now every result item has it's attributes inline. Also, operations that can be performed with the result item are in-place (for example, book a table in a restaurant or listen to a track), the result is ready to consume.

Also, SERP now has more information to explain the result. Traditional Web search engines provide the result based on some complex statistically built formula, and the result ranking cannot be reasonably explained. In our case, we have meaningful connections, as for example, *"this place is recommended for you because three of your friends have been here"*, or *"this concert might be interesting for you because you liked the performer on Facebook"*.

## 2.2 Natural language search interface

The data is well-structured and every object has multiple attributes. Objects are connected via friends' actions performed upon them. Intuitively, traditional keyword-based language doesn't fit here, we would rather search by attributes. But visual representation of searchable fields leads to overloaded interfaces with a lot of text input fields and checkboxes (take, for example, Yelp native app).

Natural language interface seems to be a good candidate to express queries for this kind of data. In natural language it is easy to mention the required fields and connections. For example, *"Show me bars in Oakland visited by MG Siegler"*.

Natural language interfaces to complex data has been extensively studied before, in late 70-ies and early 80-ies. Extensive research was done at SRI International [8] that later leaded to Siri [7], an intelligent personal assistant acquired by Apple and integrated into iPhone 4S.

## 2.3 Grouping data to build coherent stories

Each data item is like a real-time elementary signal coming from the user's friend (*"your friend has just checked in at a bar"*). The user is not interested in every check-in their friends do. But they might be interested in a fact that their friends go to this bar every Friday. That is, to make sense of these data, the challenge is to build a coherent story. *Computational storytelling* aims to provide techniques to build a story from an elementary data items, with natural language generation and building visual composition. With the rise of various data sources and data science, computational storytelling is gaining interest among the research community [6, 10].

## 3. REFERENCES

[1] A year after launch, U.S. Spotify users have listened to 13B songs, shared 27.8M. `http://venturebeat.com/2012/07/21/a-year-after-launch-u-s-spotify-users-have-listened-to-13b-songs-shared-27-8m/`.

[2] Facebook says that 1 billion pieces of content is shared via Open Graph daily. `http://thenextweb.com/facebook/2012/07/26/facebook-says-that-1-billion-pieces-of-content-is-shared-via-open-graph-daily/`.

[3] Instagram 3.0 Press Center. `http://instagram.com/press/`.

[4] Open Graph. `http://developers.facebook.com/docs/beta/opengraph/`.

[5] Twitter hits 400 million tweets per day, mostly mobile. `http://news.cnet.com/8301-1023_3-57448388-93/twitter-hits-400-million-tweets-per-day-mostly-mobile/`.

[6] D. S. A. Boyang Li, Stephen Lee-Urban and M. O. Riedl. Automatically learning to tell stories about social situations from the crowd. *8th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment Doctoral Consortium*, 2012.

[7] D. Guzzoni, C. Baur, and A. Cheyer. Modeling human-agent interaction with active ontologies. In *AAAI Spring Symposium: Interaction Challenges for Intelligent Assistants*, pages 52–59. AAAI, 2007.

[8] G. G. Hendrix, E. D. Sacerdoti, D. Sagalowicz, and J. Slocum. Developing a natural language interface to complex data. *ACM Trans. Database Syst.*, 3(2):105–147, June 1978.

[9] L. R. Janna Anderson. The Future of Apps and Web. *Pew Internet & American Life Project*, 2012.

[10] N. McIntyre and M. Lapata. Learning to tell tales: A data-driven approach to story generation. In K.-Y. Su, J. Su, and J. Wiebe, editors, *ACL/AFNLP*, pages 217–225. The Association for Computer Linguistics, 2009.