InterSocialDB: An Infrastructure for Managing Social Data

Dimitris Souravlias, Georgia Koloniari, Evaggelia Pitoura

D.M.O.D.
Distributed Management Of Data Laboratory
www.dmod.eu

Introduction

Huge amounts of social data is generated by users of Online Social Networks (OSNs)

Social Data is not in any unified structured format

- Tweets are augmented with links, hashtags etc
- Facebook pages contain images, videos etc
- Foursquare stores check-ins of venues

In this work, we propose an infrastructure for providing:

- Storage and
- Processing functionality

to applications that target at analyzing social data

Structure

- InterSocialDB Framework
- Storing social data
- On-going work
- Conclusions

The InterSocialDB Framework

This infrastructure is composed of:

- A data acquisition component that:
  - collects,
  - preprocesses,
  - models,
  - aggregates and
  - stores social data
- A data processing component that:
  - performs various analytical tasks on social data and
  - presents the results of this analysis to the user

The InterSocialDB Framework

The data acquisition component is partitioned into three phases:

- Collection

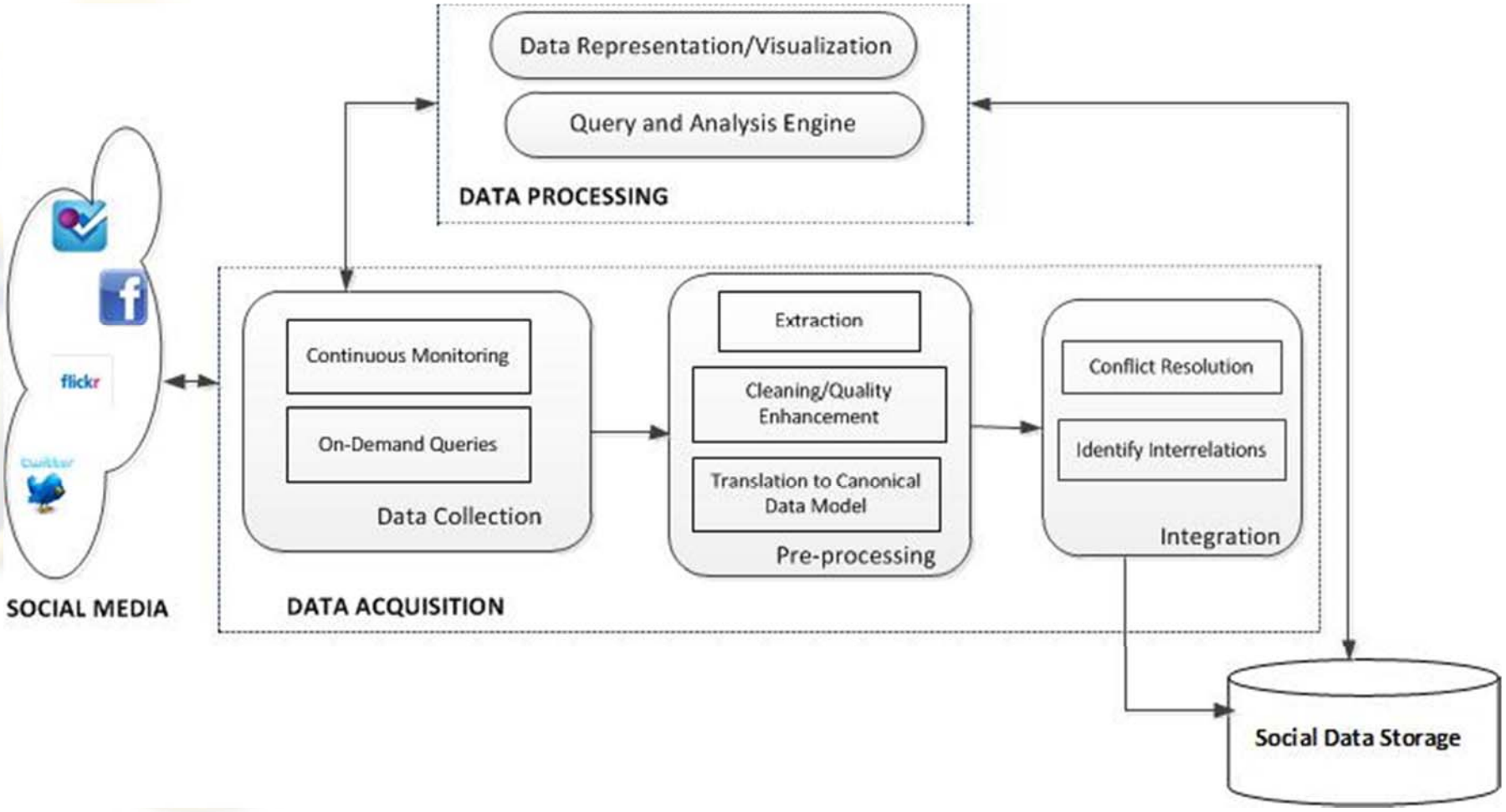
Data are gathered from social networking sites

- Preprocessing
  - extracting related information
  - data cleaning
  - translating data to a common data model

- Integration

As data are generated from more than one social networking site, we need to identify references to the same object

The InterSocialDB Framework



Structure

- InterSocialDB Framework
- Storing social data
- On-going work
- Conclusions

Storing Social Data

We focus on storing Social Graph data

There is not a widely accepted data model for storing social data

Alternatives for storing social data:

- relational database systems
- key-value stores
- document databases
- native graph databases

Relational database systems

The Social Graph can be stored in traditional relational database systems

Pros:

- Structured Query Languages (MySQL)
- ACID properties

Cons:

- Costly JOIN operations

Examples:

- Sun MySQL
- Oracle SQL

Key-value stores

Social data is stored as key-value pairs

Pros:

- Scalability
- Performance (efficient read/write operations)

Cons:

- Limited to key-oriented queries

Examples:

- HBase
- Redis

Document databases

Social Data is stored into structured formats (e.g. JSON, BSON)

Each document is associated with a document id

Pros:

- Secondary indexes on words of each document
- Social data generated by APIs have usually JSON (or similar) format.

Cons:

- No support for ACID operations

Examples:

- CouchDB
- MongoDB

Native graph databases

They store the Social Graph as it is

Pros:

- graph-oriented storage
- good performance for graph queries (e.g. shortest path queries)

Cons

- bad performance for aggregate queries (worse than Relational DBs)
- distribution issues: How can we distribute a graph database among a number of network nodes?

Examples:

- Neo4j
- AlegroDB

Structure

- InterSocialDB Framework
- Storing social data
- On-going work
- Conclusions

On-going work

Currently, we are working on:

- collecting data generated by popular social networking sites
- evaluating different data models for storing social data
- exploring temporal dimensions of social data

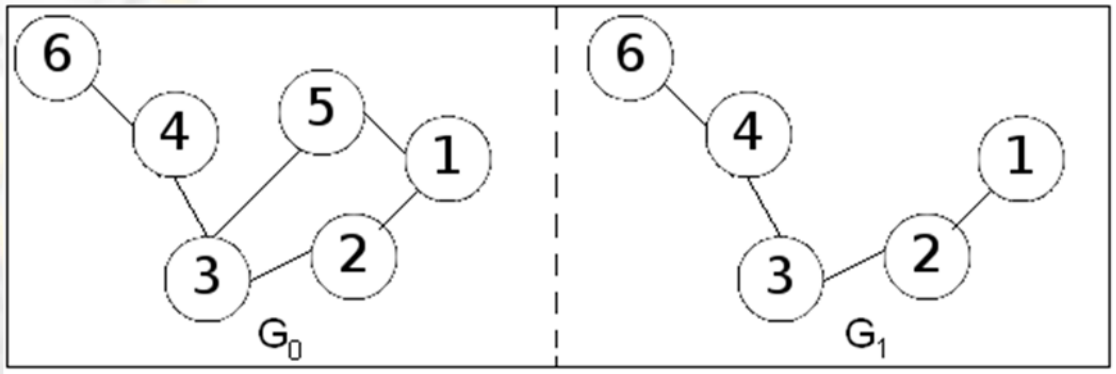
On going work

Exploring temporal aspects of social data:

- We consider live streams of social data
- Data is stored in a graph-oriented database (Neo4j)
- We propose and evaluate algorithmic methods to answer effectively and efficiently temporal queries on data of the Social Graph
  - Which is the number of Antonio Di Natale' s Facebook friends on 10th June 2012?
  - How many Foursquare users have checked-in Achaia Beach Hotel from 1st June to 1st July 2012?
  - What is the size of the diameter of the social graph of greek Twitter users on 17th June 2012?

On going work

Storing successive snapshots of the Social Graph:



2 successive snapshots of graph G, at time t_0 and at time t_1 respectively

Their difference is the set of operations:
D = { remove_node (5), remove_edge (5,1), remove_edge (5,3) }

We call this set a Delta set

On going work

Interesting questions:

- Which snapshots of the social graph to store in order to answer temporal queries efficiently?
  - storing a set of periodic snapshots and a set of deltas
  - storing only a current snapshot of the social graph and a set of deltas
- Is there an efficient way to store deltas between snapshots?
  - log file that contains a set of operations in which successive snapshots differ
  - Storing deltas as a graph?

Structure

- InterSocialDB Framework
- Storing social data
- On-going work
- Conclusions

Conclusions

Besides the analysis of Social Data, there are also a variety of critical data management tasks:

- Data Modeling
- Data Integration
- Data Storage & Indexing

We have presented the architecture of IntersocialDB, an infrastructure for managing social data

We are in the process of implementing the related infrastructure

D.M.O.D.

Distributed Management Of Data Laboratory

www.dmod.eu

Thank you!

www.cs.uoi.gr/~dsouravl

www.gr.linkedin.com/in/souravlias

References

A. Y. Halevy, *Towards an ecosystem of structured data on the web*, EDBT 2012

Y. Simmhan, B. Plale and D. Gannon, *A survey of data provenance in e-sciences*, SIGMOD Record 34(3), 2005

N. Ruflin, H. Burkhart, S. Rizzotti, *Social-Data Storage Systems*, DBSocial 2011