Towards Social Media as a Data Source for Opportunistic Sensor Networking

James Meneghello[1] **Kevin Lee**[1] **Nik Thompson**[1]

[1] School of Engineering and Information Technology
Murdoch University,
Perth, Western Australia 6150,
Email: j.meneghello@murdoch.edu.au, k.lee@murdoch.edu.au, n.thompson@murdoch.edu.au

Abstract

The quality and diversity of available data sources has a large impact on the potential for sensor networks to support rich applications. The high cost and narrow focus of new sensor network deployments has led to a search for diverse, global data sources to support more varied sensor network applications. Social networks are culturally and geographically diverse, and consist of large amounts of rich data from users. This provides a unique opportunity for existing social networks to be leveraged as data sources. Using social media as a data source poses significant challenges. These include the large volume of available data, the associated difficulty in isolating relevant data sources and the lack of a universal data format for social networks. Integrating social and other data sources for use in sensor networking applications requires a cohesive framework, including data sourcing, collection, cleaning, integration, aggregation and querying techniques. While similar frameworks exist, they require long-term collection of all social media data for aggregation, requiring large infrastructure outlays. This paper presents a novel framework which is able to source social data, integrate it into a common format and perform querying operations without the high level of resource requirements of existing solutions. Framework components are fully extensible, allowing for the addition of new data sources as well as the extension of query functionality to support sensor networking applications. This framework provides a consistent, reliable querying interface to existing social media assets for use in sensor networking applications and experiments - without the cost or complexity of establishing new sensor network deployments.

Keywords: data sourcing, data mining, data integration, social media, social data, integration framework, sensor networking, resource preservation

1 Introduction

Sensor networks are a grid of spatially distributed multifunction sensor nodes designed to cooperatively monitor environmental conditions and pass collected data in a collaborative fashion for further analysis. These networks have provided numerous benefits to society since their development, and have previously been deployed in areas to monitor volcano activity, floodplains (Hughes et al. 2008), bushfires and earthquake risk zones (Akyildiz et al. 2002). Human-oriented sensor networks (such as Body Area Networks (Chen et al. 2011)) have also been developed with the aim of providing benefit to individuals through better monitoring of health conditions, as well as epidemic detection for viruses such as influenza (Okada et al. 2009). These systems rely on analysis of data collected by the sensors in order to provide actionable advice, and the majority of these systems have application-specific implementations. There is some repurposing of collected data, but sensor networks for generic use are generally not deployed as acquiring funding without a well-defined scope is difficult.

Sensor networking applications are constrained by available data sources. Supporting new applications often requires deployments of sensor hardware that are able to collect new types of data. This data can then be used in new analytical applications or used to extend and enhance existing applications. Using existing data sources to enhance sensor networking is an attractive prospect, as it does not require the expense of developing, implementing and maintaining a new sensor deployment.

To support the extension of sensor networks in this manner, existing networks capable of providing data must be located. A candidate network should be semantically and structurally similar to sensor networks to allow a smooth application of existing sensor networking techniques to the new data sources, and also support integration with existing sensor networks. It is likely that multiple data sources properly integrated into a cohesive dataset of a standardised format could meet these conditions, as long as the data was procured in a way similar to sensor networks: a node collects events from the environment and communicates them to a central host.

Social media is an ideal candidate for use as a data source in sensor networking. It is comprised of a number of isolated networks covering much of the global population, providing penetrative reach into diverse communities. It produces very large amounts of variable-quality data with accompanying metadata. When properly filtered and cleaned, it can be a source of good-quality, relevant data for integration into sensor networking applications. Social media also follows similar design patterns to sensor networking, being primarily event-based, and can conceptually be treated much the same way as a traditional sensor network. Social users (nodes) write comments (events) that are transmitted to

followers (connections). By identifying points of interconnection between social networks and sensor networks and providing appropriate metadata, we can inject social user data into sensor networks to emulate additional sensor nodes.

The near-ubiquitous acceptance of social media within modern society has led to strong userbase growth across all platforms, with 56% of Americans across all age groups now using at least one social network and 96% of those aged between 18-35 (FormulaPR 2011, Edison 2012). Globally, 26% of the population use social media, including 57% of Australians and 46% of Chinese (Singapore 2014). Social media users use tend to use different platforms depending on their nationality - Cyworld is estimated to host 50% of the social networking profiles of South Korean users, while Mixi is more popular in Japan and 20% of Orkut's population is Indian (Vasalou et al. 2010). Integrating these social networks would provide a truly global, culturally-diverse, rapidly-updated data source for sensor networking.

There is great demand for this social data within the academic and industrial communities. Social studies rely heavily on qualitative results from user-generated content such as surveys, while advertising entities harvest user metadata to more appropriately target advertising. Currently, data for these activities is collected directly from users, either by asking for their participation in studies or by using web-based tracking technology to monitor their use of the internet. These two forms of data collection have also been implemented in social sensor networks, as participatory (Burke et al. 2006) and opportunistic (Lane et al. 2008) sensing. Problematically, both of these implementations require ongoing software deployments and usage of resources to monitor the environment - potentially dissuading users from continuing participation.

Using social media as a data source for sensor networking presents a number of challenges, from realistic and technical perspectives. Allowing the mass collection of data from users in an easily-queryable format could have unintended consequences, such as the mass-reporting of user locations (Borsboom et al. 2010). Analysis is made more resource-intensive by the high throughput of data posted to social media - Facebook alone collects and warehouses almost half a petabyte of data per day (Ching et al. 2012). Platforms to aggregate social media already exist, such as Datasift (DataSift 2010) and Gnip (Gnip 2008), and provide this service by performing widespread collection of all social media data, requiring substantial storage and processing infrastructure. These platforms are not developed with sensor networking integration in mind, and the event-based behaviour of social nodes is lost during the integration process.

In this paper, we propose a framework which performs data mining, filtering, integration and querying of generic social networks. This framework is designed to integrate with sensor networking systems and analytical tools to allow researchers and industry to leverage user-driven content for the purposes of event-detection, metadata analysis and general user analysis. This is achieved through conflating the concepts of social networks and sensor networks and provides a robust data integration chain with a data querying engine designed to streamline access to desired data. In addition, tools for mining data from social media are detailed and integrated into the framework to provide a more complete picture of each user and their content than merely parsing a single Application Programming Interface (API). The framework provides a simplified query interface to a cohesive data-set made from generic social media networks, providing researchers and industry with access to a large, global, generic set of user-generated metadata and content.

Section 2 presents a review of literature and previous work completed in the areas of social data mining and opportunistic sensing. Section 3 analyses the use of social media as a data source for sensor networking, as well as the challenges involved in doing so. Section 4 describes a framework designed to alleviate some of these challenges by optimising the data collection process. Section 5 evaluates the data sourcing algorithm ability to discover potential data sources relevant to a given topic. Finally, Section 6 presents some conclusions.

2 Background

The process of collecting data from disparate social networks and integrating it for common use utilises techniques from a broad spectrum of computer science research. Supplying user data as a data source for sensor networking is commonly referred to as participatory or opportunistic sensing, where participatory involves the direct contribution of data from users where opportunistic uses passive observation of users. This section examines the use of both of these methods of data collection from social networks, as well as examining previous applications of social data in analytical studies. Finally, data integration techniques are examined for use in linking collected data across social networks.

2.1 Participatory and Opportunistic Sensing

Sensor networks are typically a spatially-distributed mesh of potentially thousands of sensors, with sensor nodes deployed in a configuration befitting the desired application. Each node communicates collaboratively in order to move collected data back to the sink node - using other nodes as a relay to increase communications range. The sink node then typically has a connection back to more powerful resources for processing the information, though this may also be collected manually. Low-power wireless sensor networks are often used to collect data from the natural environment, the human body, mechanical equipment and many other sources. This data is, in turn, analysed and logged or used to actionable effect - for example, to trigger a warning in the event of an earthquake.

Participatory sensing uses existing mobile devices and users as nodes within a sensor network (Burke et al. 2006) by encouraging users to gather, analyse and share local knowledge in what is commonly referred to as "crowdsourcing" (Kanhere 2011). By treating users as sensor nodes, participatory sensing takes advantage of resources that already exist to extend the reach of sensor networks for a number of purposes, including urban planning and policy development. Smartphones also come with a number of sensors that can be made available for participatory sensor networks, including the important contextual metadata sensors of location and time.

Participatory sensing requires direct and active participation of users, which comes with a number of challenges. One of the key issues with using people as nodes in sensor networks is that they are often less reliable than hardware sensors (Hughes et al.

2014). Users are able to choose whether to provide data on a requested topic, and may choose not to do so. Participatory sensing is named as such for a reason - without active participation, the system is unable to produce useful outcomes.

Further research in using participatory sensing for urban sensor networks has resulted in the development of several auxiliary approaches. Opportunistic sensing reduces the burden on the user by lessening direct participation - instead relying primarily on the devices the user carries around (Lane et al. 2008). Applications on smartphones can take sensor measurements without bothering the user and pass it along to sink nodes over mobile networks. While quality data is still dependent on the user's presence in an area of interest (Min et al. 2013), the user is no longer required to directly answer queries. Data produced by hardware sensors through opportunistic sensing is objective and usually of good quality, as it is not provided directly by the user and is in a standard format (Campbell et al. 2006). Under this approach, collected data is only provided by sensors directly attached to the device, and users cannot be queried for additional information. This use of smartphones in opportunistic sensing is commonly referred to as Mobile CrowdSensing (Ganti et al. 2011).

In order to leverage opportunities provided by both participatory and opportunistic sensing, some architectures combine both techniques (Guo et al. 2014). This allows users to provide data as requested and fill contextual data using opportunistic sensing. Concerns with data quality arising from direct user involvement remain.

A major problem with many implementations of both participatory and opportunistic sensor systems is the requirement of manual application installation by the user. To retrieve data from a smartphone's sensors, an application must be installed that allows the smartphone to operate as a sensor node. These applications are usually neither large nor difficult to install, but even the smallest of hurdles can hamper efforts to leverage users as sensors. These services also consume energy on devices that have limited resources, which may drive some users away. Use of these services usually relies on providing the users with some kind of benefit, which is an ongoing cost.

To alleviate these issues, sensor middleware suites have been developed (Hachem et al. 2013, Hughes et al. 2009) for smartphones that only require a single installation, even if there are multiple different deployments using the device. These frameworks vastly simplify sensor installation and reconfiguration, further reducing burden on the user.

The framework proposed in this paper aims to use indirect participatory and opportunistic sensing at a higher level by analysing social media. While facing similar data quality challenges to both techniques, we work around user participation by only monitoring existing social media usage. No additional applications need to be installed and no extra resources are consumed by personal devices.

2.2 Social Media

Effectively deriving useful data points from social media is a topic of much discussion, and presents a number of challenges (Maynard et al. 2012). Posts by users on social media are usually free-form - the data comes in no particular standard format, as they often represent part of a stream of consciousness from a user. Additionally, posts are not limited to text and users often share pictures, videos, diagrams or graphs which present unique challenges to data analysis.

The proliferation of social media has driven content and context creation, but has also made it more difficult to locate appropriate data sources (Wandhöfer et al. 2012). Where topics were once discussed in semi-centralised locations such as Usenet, the integration of commenting systems into many different news and blog sites has dispersed information to this point where it is unrealistic to attempt collection of all relevant discussion relating to a topic. Instead, discussion centers can be discovered by tracking the spread of topics across the internet and monitoring the most active communities.

Many existing studies on using social media for event detection tend to focus on a single social network (Li & Cardie 2013, Cameron et al. 2012, Sakaki et al. 2010, Robinson et al. 2013), limiting collection specifically to the demographics represented on the chosen platform. Cultural demographics can vary significantly per platform - Cyworld is estimated to host 50% of the social networking profiles of South Korean users, while Mixi is more popular in Japan and 20% of Orkut's population is Indian (Vasalou et al. 2010). In order to develop a truly global and cross-cultural social media monitoring system, data collection mechanisms should be extensible to any form of online social media, rather than a specific few platforms.

In order to realise the concept of global generic social sensor networking, a diverse set of data collection and integration techniques need to be utilised. Different social networking platforms operate on different data structures, using different storage engines and producing entirely different output. Even considering this, social media data is conceptually similar across platforms, consisting of a number of common constructs (users, friends, connections, messages, events). Because of this conceptual similarity, it is possible to integrate this data into a single queryable dataset through the use of data collection and integration techniques.

2.3 Generic Collection and Integration

Generically utilising data from different web services requires integration between social media platforms. While programmatic access to web services has vastly improved since the introduction of Web 2.0 paradigms, interoperability and data integration between social networks remains an issue. There is limited interoperability of services provided for the purposes of open authentication, but content sharing is usually limited. There have been some attempts to apply semantic web principles to the problem, including Semantically-Interlinked Online Communities (SIOC) (Breslin et al. 2009) which describes social networks using the Resource Description Framework (RDF) to improve interoperability. While RDF has yet to reach widespread adoption and is unavailable for use with many social media systems, the data structures and principles in use provide a good platform upon which to base further work.

There are a number of challenges surrounding the matching of social media profiles between networks. The FOAF (Friend-of-a-Friend) Project (Brickley & Miller 2000) attempts to extend Semantic Web efforts to social media by providing a base for user profile matching across networks. It does this by combining names and user metadata (such as

location, email addresses or education details) to provide a more substantial set of data to improve accuracy of matches. As with the SIOC project, few social networks actively support such efforts and most do not provide FOAF output for users.

The majority of studies conducted using data collected from social media follow a reasonably similar process: conduct (manual or automated) searches of a social network, save or export returned data in a simple format, and perform analysis (online or offline) to determine answers to a particular query. While this process is reasonably generic, there have been no attempts to develop a querying engine for social media analysis. Query engines have been developed for a range of other purposes (Khoury et al. 2010, Madden et al. 2005) with the intention of providing a standard querying interface and abstraction layer on top of complex datasets. The development of a query engine that can genericise collection and analysis over multiple social media platforms without requiring large infrastructure outlays would be a useful addition to social media research.

2.4 Integration of Social Data

Data integration can be a complex process, depending on the complexity, relevancy and size of the converging datasets. Numerous techniques exist to handle the process of querying integrated datasets, designed with different operating requirements. Some techniques require the offline transformation and storage of data for later querying, while others can handle-queries in real-time by inferring the goal of the query and transforming appropriate data as required.

Extract-Transform-Load (ETL) systems are commonly used in data warehousing, where data is initially cleaned and transformed for storage and later use (Vassiliadis 2009). Due to the extensive processing that data undergoes in order to be in a clean state with unified schema, this process is generally not real-time, and can suffer from data freshness issues. This approach is therefore only useful for use in delayed queries, and its use with social media would exclude any real-time sensing and reactive systems.

There are also dataflow processing systems such as Google Cloud DataFlow (Perry 2014) that would be appropriate for the task of integrating large social datasets. In conjunction with the use of BigQuery (Sato 2012), we can provide a platform providing a queryable interface to real-time integration of social streams that can be use in decision support systems, providing actionable insights.

Some attempts have previously been made to support interoperability between social networks, particularly the Semantically-Interlinked Online Communities (SIOC) project (Breslin et al. 2009). The SIOC project uses the Resource Description Framework specification to integrate some elements of social networks, particularly the association of user profiles over different networks. SIOC exporters have been developed for a limited number of social networks.

Commercial services such as Datasift (DataSift 2010) and Gnip (Gnip 2008) perform integration and long-term storage of social data. The integrated data is then presented to applications through a query API, which is often used by companies to monitor their online social profiles. This allows for rapid response to user complaints directed at social followers (rather than the company itself, through a complaints procedure). The integration process of these systems does not consider the event-based behaviour of social nodes, and are generally unsuitable for use in extending sensor networking. Ongoing access costs can also be a constraining factor for applications intending to use social data, and independently deploying such a system requires an infeasiable level of processing and storage infrastructure for most projects.

3 Social Media as a Data Source

Social media is comprised of a number of isolated networks covering much of the global population. They are primarily used for facilitating communications between people over the internet, social networks have also been used to gauge user response to advertising (Taylor et al. 2011) and also as transmission mediums for sensor networks. Potential sensor networking applications rely on available data sources, and the penetrative reach of social media into global communities and vast amount of data posted provides an ideal data source for this purpose. Properly prepared, social data is suited for further analysis, particularly for detecting cross-cultural events or insights. Using social media as a data source for sensor networking is not a trivial task. There are significant challenges facing any platform aiming to facilitate this dataflow.

To use social media in this way requires the integration of disparate social networking platforms. There have been attempts to ease the bidirectional migration of data on social networking platforms, as noted in Section 2.3, but these have generally had little industry support. Most networks have an interest in "locking-in" customers, to dissuade them from migrating between networks and losing associated advertising revenue. Easing data integration processes would also accelerate migration between competing networks, so efforts to standardise export formats are unlikely to ever achieve industry co-operation. Therefore, new techniques supporting inter-network social data integration need to be developed in order to facilitate the integration of social media into sensor networking.

To enable the use of social media as a data source for sensor networking, a framework supporting this goal must be developed. This framework is made from a number of processes designed to take heterogeneous data sources, integrate them into a common schema, provide generic querying functionality and present appropriate output for further analysis or integration into sensor networks.

This framework must address a number of key challenges:

1. Collecting data from all available sources for use in sensor networks can be infeasible due to the sheer amount of data being produced. To query social media data in an efficient manner while retaining the ability to query as much relevant data as possible, some way of reducing the incoming flow of data to exclude irrelevant sources is required.

2. To use social media as a sensor, sourced data needs to be retrieved. Retrieving social media data can be straightforward if an API is provided, or require manual scraping if key information is not provided by the API.

3. Social media data is noisy because it is almost entirely user-generated and doesn't adhere to a standard structure or format. This data requires extensive cleaning to remove spam and bring low-quality content up to a usable standard (Agichtein et al. 2008). Without performing this cleaning, using the data in automated applications becomes significantly more difficult.

4. In order to present the collected data in a format that can be integrated with sensor networking applications, there must be a way comparing diverse datasets. Without a method of mediating schematic differences between data sources, every application would require manual mapping of data structures.

5. The execution of queries over datasets as large as those that social media provide can be challenging, due to resource constraints and missing values. Sensor networks also deal with data in many different ways and can require extensive pre-processing and aggregation to be performed prior to use.

6. Some sensor networking applications can integrate data in simple formats such as JSON, but others require more complex techniques (such as event injection) to emulate social nodes as sensor nodes.

In order to properly define steps in the social integration process, each of these challenges is represented by a key area of functionality, respectively: Sourcing, Collection, Cleaning, Integration, Querying and Presentation. Once a solution for each of these challenges has been identified, they can be joined into a unified process supporting the integration of social media and sensor networking. These challenges are described more fully in the sections below.

3.1 Sourcing

One of the most simple optimisation steps in large-scale data processing applications is to filter incoming data, resulting in reduced processing for each successive step in the integration process. As social media has an extremely broad scope, relevant data sources must be located in order to efficiently leverage social data in sensor networking applications. Without this initial filtering process, much of the social data processed may be completely irrelevant to the application. As this data must undergo cleaning, integration, storage and querying, early filtering can result in significant resource savings. Hence, the process of finding quality data sources is very important.

Sourcing involves locating social data streams that provide data relevant to the intended application. On the internet, many of these sources can provide access to historical or real-time data streams. Both of these types of data can be useful to extend or enhance sensor networking applications. Real-time data can actively replace or enhance sensor nodes with additional data sources, while historical data can provide longitudinal context. Data sources can also be normal rich-text web pages that can require substantial parsing and cleaning before use.

Most of these data sources also provide access to further sources. Social media posts often contain hyperlinks to static web content, and also contain metatags (such as other usernames and hashtags). Static content is often interlinked, with most websites providing hyperlinks to other relevant material on associated sites.

Figure 1 presents the process by which sourcing occurs. The query specifies relevant keywords, which can be expanded upon by use of predefined databases and appended to by examining oft-used keywords on strongly-relevant search results. These keywords are used to query known search APIs, returning a list of locations that potentially contain results. The exact method used for data access and searching can vary for each system, as each platform provides differing methods of accessing and filtering data. Some examples of different methods are:

Facebook Using the API, search for public posts with related search terms, popular news feeds and other items of interest. Additionally collect comment authors.

Twitter Using the API, search for public tweets containing related keywords. Additionally collect information about replies to tweets, and also examine hashtags commonly appearing within the initial set.

Blogs Using Google's Search API, search for public blog posts containing relevant keywords, including author information and comments. Additionally collect relevant results from commenters' own blogs.

Forums Using manual page scraping and authentication for forum software such as VBulletin and phpBB, search for forums containing threads relevant to our query.

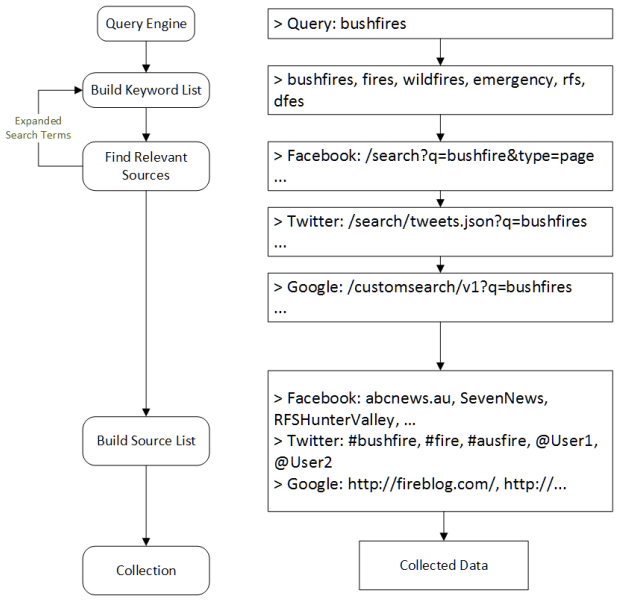


Figure 1: Finding relevant sources with the sourcing process

Upon completion of this initial phase, a list of relevant places to look for information relating to this single query has been collected. Not all data present within these sources is useful, and some may be completely irrelevant. By selecting a fairly wide sub-set of available data, we have already limited the initial collection to a feasible scope, allowing for more accurate filtering later. This optimisation can potentially exclude obscure results, but this is a necessary trade-off.

3.2 Collection

To use data from identified sources in applications, data must be collected. There are three main approaches to data collection: the use of Application Programming Interfaces (APIs), access to formatted real-time streams, and ad-hoc data collection.

Using APIs to collect data involves sending a specifically-constructed request to a provided server. This request contains a number of parameters used to focus the scope and range of data returned by the API. The requested data is then returned to the requester in a well-defined format. Some APIs provide a limited view of data available to the source, requiring the requester to follow-up with an ad-hoc request for more data. These initial API requests often lead to further queries for related data (such as collecting user profile data for users that have posted in a comment thread).

Data can be collected from real-time streams by requesting stream access from a source server. The source server then pushes a constant stream of real-time data towards the requester, in what is often called a "firehose" stream. This data is presented in a well-defined format that can be mapped to an appropriate data schema. Many firehose streams consist of all available real-time data being produced by the source. The amount of data provided by this real-time stream can be problematic for systems operating with restricted bandwidth, processing or storage resources, and can easily overwhelm low-resource systems.

Ad-hoc parsing or scraping of data can be used for data sources that have no defined format and do not provide an API or formatted data stream. This usually requires the manual development of parsing algorithms tailored to specific sources. Automated parsing algorithms can also be used, to varying levels of success. Ad-hoc scraping can also be used to enhance data provided by APIs, where data is missing or deliberately restricted from API access.

Figure 2 illustrates an example structure for handling data collection across multiple source types and authentication methods. For many sources, collection can occur through accessing a network-provided API, such as Facebook or Twitter's APIs. Many APIs operate on similar authentication standards (such as OAuth (Hardt 2012), depicted), requiring minimal work to write wrappers for a generic scraping engine even across a wide variety of different sources. Other sources require collection through page-scraping, a more resource-intensive process that involves writing parsers for source pages and handling page authentication in a customised manner. While API access is generally less resource-intensive and requires less development work than scraping, scraping can potentially provide more data.

Data collection is limited by the interfaces provided by the API, e.g. the Facebook API does not provide location information for users, even if the data is set to be publically viewable. A second-pass scraper can access the profile page directly to collect this information, providing more complete meta-data but is also more resource-intensive. Second-pass collection can also involve further queries to the API for related data (such as collecting user profile data for users that have posted in a comment thread).

The collection process can be very easily distributed across resources, as each first-pass operation is isolated. The initial source list can be packaged by a workflow manager and work assigned

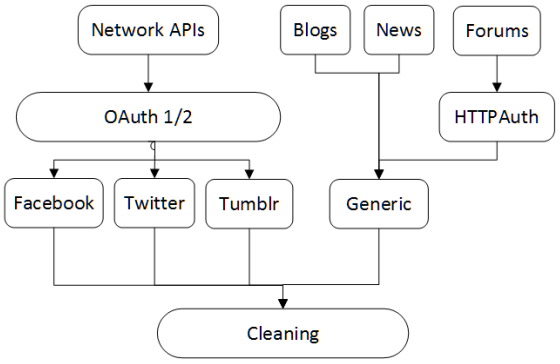


Figure 2: Collection module flow

in an optimised manner to ensure that requests are spread evenly over workers. When workers have run out of allocated work, second-pass collection can occur in a manner that ensures atomicity.

3.3 Cleaning

Data collected from social media and other internet sources is noisy, and can require extensive cleaning and processing. Most social data is user generated and adheres to no particular structure, primarily being unstructured conversational language. Differences in data formats between sources can also be problematic, such as the use of different date standards between cultures and systems. Conceptual differences between individual data elements across different networks can also require manipulation of data into a unified standard.

There can also be structural problems with social data depending on its age and nature. Legacy data has often undergone repeated format and schema shifts, so data from these sources can require extensive cleaning. Modern sources adhere to stricter standards and usually require less cleaning.

The method of collection can determine the extent of data cleaning necessary. API-provided results are usually retrieved from the database and output without presentation, and therefore adhere to internal database quality standards. Data returned from ad-hoc scrapers can require extensive cleaning, including removal of undesirable elements such as HTML tags and extraneous characters.

It is at this point that cleaning data for privacy reasons may be handled. Anonymisation of users can be performed during the cleaning process to ensure user privacy for applications that do not require identifying information to be stored.

3.4 Integration

Data from different social sources are collected using their original schemas. Attempting to query data across these many schemas can be difficult without proper data integration, as it requires the query engine to specifically deal with many different data formats and sources. Data integration provides the ability to query data over a mediated schema, in which all sources can be treated in a generic manner. To use this data set in sensor networking applications, collected data must be integrated.

There are a number of available data integration techniques supporting this goal, including those discussed in Section 2.4. The integration process can be operated by predefined mappings from collected data schema to the global schema, such as the RDF

specifications used in the SIOC project (Breslin et al. 2009). It is possible to use automated mapping algorithms (Doan et al. 2001) to develop page and API wrappers that require minimal modification by a developer, reducing development time.

These mappings can be applied to data in a number of different ways, depending on the database engine used. Data mappings to relational designs can store the integrated data in relational database management systems, while there are also options for NoSQL, key:value and tuple stores. Each provides the ability to perform a different form of integration, so any integration design process should also take the choice of query engine into account.

3.5 Querying

To provide only relevant and desirable data to sensor networking applications, there must be a method of restricting, aggregating and analysing integrated data. Analysis performed may require results aggregated by categorical variables, and this can be handled using querying. Querying is an important part of data analysis, and is readily available in all database systems.

As the integrated data is loaded into a relational database management system with a unified schema, query processing and optimisation is greatly simplified. Additional operators and aggregate functions can be added to the query syntax to allow for the sourcing and collection mechanisms detailed in 3.1 and 3.2 respectively, including the ability to filter by Site and restrict potential data sources by keyword. Queries can be provided in both SQL-like syntax or to a RESTful API, using JSON or XML.

The utility of integrating social data into sensor networking applications ultimately hinges on the ability to adequately query the data. Other systems designed to integrate diverse data sources into applications place significant emphasis on their querying engines. Software projects such as Google's BigQuery and Facebook's Presto were designed specifically to query large-scale sets of data in real-time. Querying social data therefore needs to be efficient in order to provide real-time results to sensor networking applications.

3.6 Presentation

Effective data presentation is required in order to use integrated data in sensor networking applications. Applications often handle different input data formats, such as XML, JSON or event packets. Presenting the data in this format for integration should be handled in such a way as to be compatible with any application.

Presentation can be delegated to client applications, with the querying engine only providing result data in a variety of text-based formats such as JSON and XML. This data can then be provided for import into other systems or directly analysed for use in graphical applications. For use with automated systems, queries can be designed to provide simplified output to directly trigger actions or provide detailed output for further use in Decision Support Systems.

Direct integration is more involved but allows for social nodes to be used directly in sensor networking applications. This works by developing output formatters that exist within those networks and rebroadcast data in the format required, often event packets. This allows social data nodes (users) to directly act as nodes within the sensor network.

4 A Framework for Supporting Social Media as a Data Source

In order to support the integration of generic social media data into sensor networking applications, the functional components described in Section 3 are implemented in a streamlined data processing framework. This framework encapsulates all necessary operations from initial data request to sourcing, collection, cleaning, integration, querying and presentation of returned results. For ease of presentation, the framework architecture is presented in multiple views.

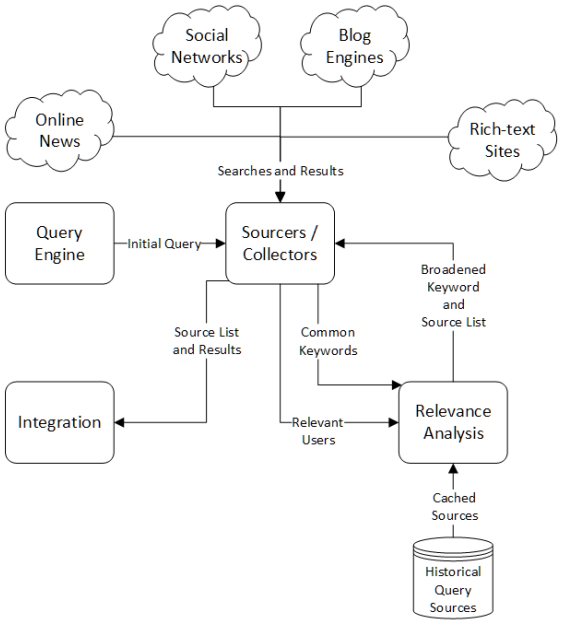


Figure 3: The process of finding new data sources

Figure 3 presents the workflow for sourcing and collection. In this stage, the framework takes the initial query and expands upon the specified keywords in an iterative process. Potential sources are taken from historical data and queried for relevance, and commonly-recurring users and keywords are crawled to discover additional relevant sources. A wide range of sources are discovered, analysed and discarded during this process to ensure adequate data coverage of relevant sources.

Sourcing is a two-step process. Using an initial seed set of keywords, the Sourcers locate sources and use content from these sources to expand and reinforce the keyword lists, as well as isolating relevant metatags (such as URLs or usernames). In the second step, the metatags are put back through the Sourcers to locate additional sources and further reinforce keyword and source lists. Each additional iteration of this process serves to further strengthen the keyword and source list, effectively using machine learning to focus the search space.

Sourcing can be performed in two modes. The first is a narrow search that attempts to find the most relevant sources while being hesitant to expand the search space. The second is a much broader search that uses all available content to train the keyword set and broaden the search space, finding potentially-relevant sources over a much larger area. Narrow searches are designed to find the most relevant sources quickly, while the broad search can find distantly relevant sources but additional resource requirements.

For organisational reasons, the sourcing components are integrated with the collection components. Functionality is shared between these two processes, with some content collection required to assess relevance of sources (and complete collection required during the collection phase).

The sourcing and collection software processes can be distributed across multiple resources, only requiring synchronisation to finalise the completed source list and return the completed collection results. Individual site wrappers may also be distributed, which can be particularly desirable in the instance of micromanaging API throttle limits that tend to differ between platforms. Distribution of these processes provides significant performance boosts in the sourcing, collection and filtering phases.

Relevant sources are cached for two purposes. The first is to expedite the sourcing process for repeated similar queries and reduce overhead and external API usage, which associates sources with particular keyword sets. The second is to identify rich-text websites that contain a large amount of useful data or pages. By keeping track of which sites are providing useful data sets, candidates for further wrapper development can be identified.

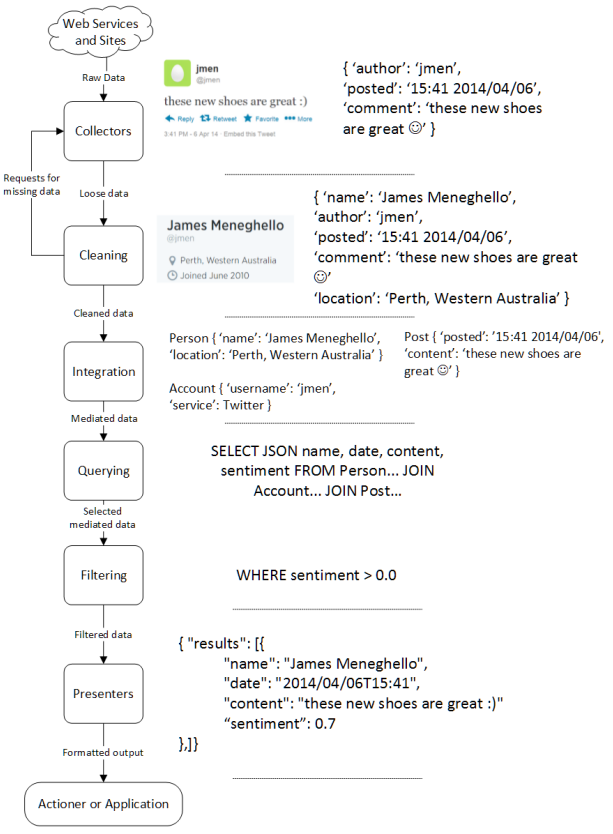


Figure 4: Example data flow within framework

The framework wraps around a Relational DataBase Management System (RDBMS), as shown in Figure 4. This wrapping occurs in multiple directions by allowing modified incoming queries, automating data collection, extending available aggregate and filtering functions and also providing for formatted output. For simplification, Figure 4 does not show processes relating to sourcing and first-pass filtering, which are instead shown in Figure 3.

After determining a source list, the framework scrapes information from sources using collectors. The collectors are either source-specific wrappers or generic web scrapers that can collect varying amounts of information from sources. Source-specific wrappers (e.g. Facebook or Twitter) return data in a standard format that can be easily integrated, while the generic scraper returns data that may be low-quality or require cleaning, as discussed in Section 3.3. This cleaning process can require further related data to be requested from sources.

The integration component takes cleaned data and integrates it over the mediated schema, as shown in Figure 5. Once complete, the data is fit to be inserted into an RDBMS using a pre-defined schema. This schema was developed using both the Semantically-Interlinked Online Communities (Breslin et al. 2009) and Friend-of-a-Friend (Brickley & Miller 2000) projects as inspiration, allowing for simple output to common semantic web formats while still retaining a high standard of performance over large datasets. The schema has been modified to work within an RDBMS, which retaining as much flexibility as possible.

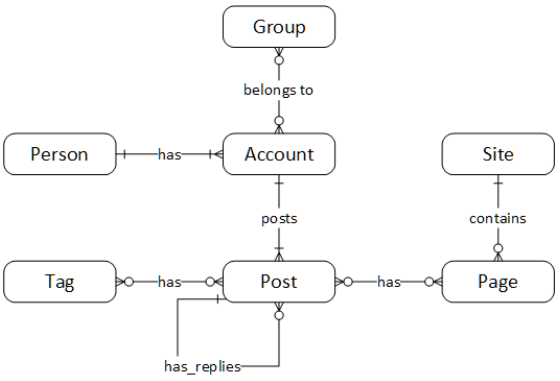


Figure 5: Internal mediated data model

Figure 5 presents the internal data model used to store integrated web data. Designed for use with relational database management systems, the model supports enough expression to adequately cover most social network and social page examples. Not all aspects of the model are utilised by every network, and some networks use certain models differently: while Twitter and Facebook can use the Tag model to represent hashtags and other metadata, blogs can use the same model to represent word tags as part of the word clouds used in most blogs.

Some model entities in Figure 5 see significant reuse due to their generic nature. Blog posts, comments, tweets, statuses and forum posts are all conceptually similar, generally consisting of a limited attribute set: title, date (posted/created), author (and other user details), content and a set of replies, along with other assorted metadata. A simple Post entity (related to Accounts to provide author and commenter data) can satisfy all of these requirements, including any comments or replies by using an adjacency relationship.

Queries submitted to the framework are translated before being passed through to the RDBMS, as there are extensions made in order to support sourcing and presentation and the original queries are not SQL. Custom second-pass filters and aggregators can be called directly, assuming the RDBMS supports user-created functions. Using the filters in this way also allows the system to cache and optimise queries natively where possible,

without requiring extensions to the database engine.

Once the query has been completed, the data can be formatted by presenters. These presenters can be in the form of a RESTful Web API that outputs JSON, XML or graphed images, or can alternatively be integrated into sensor networking systems through use of custom event generators. Normally, presentation is handled at an application-level and is inappropriate for inclusion, but the platform effectively wraps the RDBMS and provides this functionality to enable simple access to the data. Accessing RDBMSs without an interface layer raises the barrier to entry for users, requiring direct console access and usually provides data in awkwardly-formatted text.

5 Evaluation

One of the prominent and novel advantages of the proposed framework over existing approaches is the use of on-demand social data sourcing. Effective data analytics relies on the presence of quality data sources, so this is an important goal. The evaluation provided therefore emphasises data sourcing, and examines the relevance and quality of data sources discovered.

5.1 Experimental Setup

A proof of concept experiment for the sourcing algorithm was implemented in Python 3.4, taking advantage of the Natural Language ToolKit (Bird et al. 2009) library for text mining. The sourcing algorithm was executed on an Amazon EC2 m3.medium instance, and individual experiments were conducted in a single run to ensure that the available social data did not dramatically change between runs.

Sourcers were implemented for Twitter and Google Custom Search, as well as a generic web scraper used to further build keyword lists from sources. Both services are subject to API throttling limits, and these limits are taken into account during the sourcing process.

The process for each experiment consists of a series of iterations. A single iteration consists of the following steps (subject to configuration values):

1. Search Sourcers using initial seed keyword
2. Generate target source list
3. Retrieve content from list of sources
4. Mine content for commonly-occurring keywords and metatags (URLs, Usernames, Emails)
5. Further add to source and keyword lists
6. If [broad_search]: Search Sourcers using metatag lists (ie. Twitter users) and add to sources
7. Iterate using expanded keyword set

There are a number of possible configuration options for each experiment, which are explained below:

broad_search Whether the sourcing algorithm should also take sources from collected content

max_search_results The number of results to return from a Sourcer search (important to avoid API throttling)

The broad search algorithm also collects special metatags from content, such as usernames and hashtags on Twitter, and email addresses or URLs in static content. These metatags are then used to broaden the search scope, discovering additional search vectors and providing a significantly higher number of data sources while being subject to much higher noise levels. These searches are examined in the next section.

5.2 Relevance

The relevance of discovered data sources is an important metric in evaluating the usefulness of this sourcing algorithm. In order to evaluate this, the algorithm was given a broad seed keyword ("Australian politics") and let to run over multiple iterations, with an upper limit on the number of search results returned from each API of 250 for Twitter and 50 for Google. The source list was exported to a comma-separated value file and each source was manually evaluated for relevance to the topic. The algorithm was then executed twice: once as a normal search, and once with the additional broad searching options enabled.

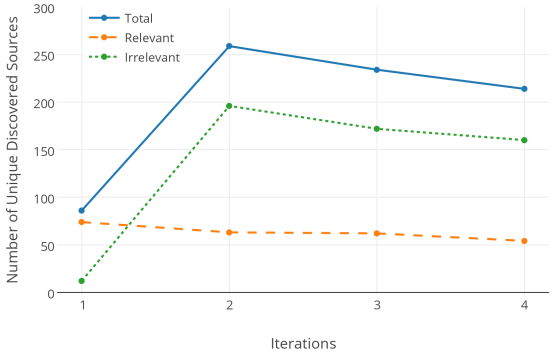
We define sources as relevant based on a number of factors: whether the source provides data related to the queried topic, the ability of the source to provide ongoing data and other sources, and the quality of data provided and its ability to be used in queries. If a source satisfies these criteria, it is considered relevant. All other sources, including those recorded as a result of algorithmic mishaps (such as misparsed hyperlinks and advertising servers), are deemed as irrelevant. Relevance is a therefore a boolean result.

As to be expected, the most relevant sources are quickly and easily found by a normal search with a low percentage of irrelevant results in the first sourcing iteration, as seen in Figure 7a. The broad search finds a higher number of relevant sources, but with a significantly higher number of irrelevant sources, shown in Figure 7b. In both instances, these sources are expanded and new keywords are derived, and additional relevant sources continue to be found at a lessening rate.

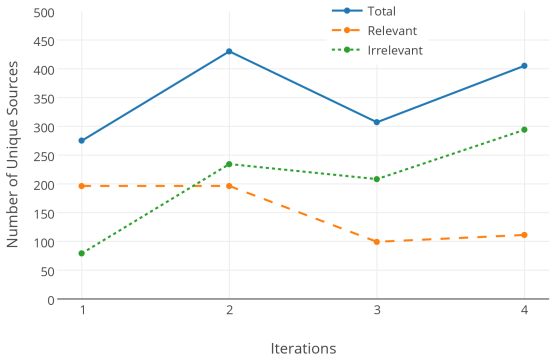
After the first two iterations, new sources continue to be found at a relatively linear rate. The relevance of discovered sources decline steadily relative to the total after the second iteration, but still maintain an acceptable rate of discovery. Figure 6b shows a steep increase in irrelevant sources during the fourth iteration, as the search space expands out beyond any semblance of relevance. The broad search maintains a much more even percentage of relevant results over the search space.

5.3 Keywords

The search keywords (initially seeded as "Australian politics") are expanded based on discovered content and the most relevant keywords float to the top of the list. The first iteration of both searches immediately expand the keyword set to contain mostly relevant keywords, as seen in Table 8a. Successive iterations narrow the search space down to a specific set of topics that quite accurately frame Australian politics. Interestingly, the broad search (which relies more heavily on page content rather than Twitter) contains a higher number of historical keywords, seen in Table 8b. A number of these keywords from a broad search relate to the government as of 2013, whereas those from the

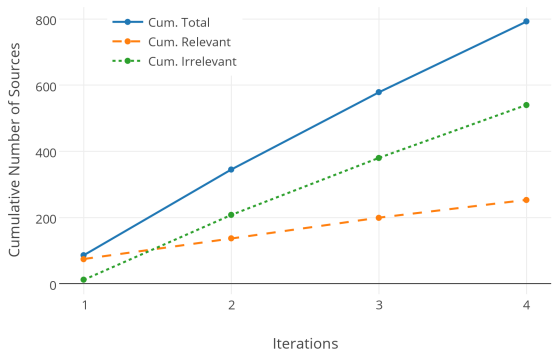


(a) Normal search

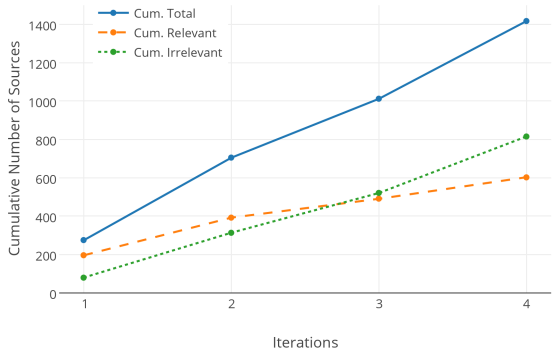


(b) Broad search

Figure 6: Relevance of data sources



(a) Normal search



(b) Broad search

Figure 7: Cumulative relevance of data sources

normal search in Table 8a relate more to the current state of Australian politics, circa 2014.

The difference in keyword sets between the search types also affects the relevancy of discovered sources after successive iterations. The iterative improvement of the search space during broad searchers potentially explains why even the third and fourth iterations of searching still contain a relatively high percentage of relevant results, as seen in Figure 6b. The normal search relies on a much smaller set of newer content from which to derive keywords, resulting in a lower discovery rate of historical data sources.

5.4 Signal-to-Noise

The signal-to-noise ratio of each search type was also examined, relative to the total number of sources discovered. A normal search discovers relevant data sources at a ratio of over 7:1 during the first iteration (Figure 9a), but immediately drops to a very low success rate in successive iterations. By comparison, the broader search starts with a much lower success rate of approximately 2.5:1 (Figure 9b), but maintains a steadier ratio well into later iterations, providing a more steady flow of new relevant sources. As explained in Section 5.3, this is likely due to the broader search touching on a larger source of historical data sources due to differences in keyword selection.

5.5 Analysis

The results of the sourcing algorithm indicate that the search methods (normal and broad) operate along different parameters. Normal searches rely more heavily on new data provided by sources such as Twitter, while the broad searches derive keywords primarily from older data sources such as newsfeeds and articles. As a result, the training of keyword sets tend toward two different trends: modern and historical, but could also indicate a disconnect in discussion between traditional media and social media. Both of these search types are useful, depending on the desired application. Overall, both search types provided a significant number of relevant data sources and ultimately achieved their goal - to optimise the collection of social media data.

There are a number of improvements that could be made to the sourcing algorithm. One would be to increase the use of training to include potential sources, preferring those new sources that had multiple existing links to discovered sources. A second involves a combination of both approaches, using historical keywords to search social media sources and modern keywords to search historical data sources.

6 Conclusion

This paper presents a framework that supports the sourcing, collection, cleaning, integration and presentation of social data for experiments and applications. The output provided by the framework can be used to drive generic sensor networking and other social analytic applications without requiring the significant hardware infrastructure investment used to store social data for later querying. This allows for the use of social media as a data source for generic sensor networking applications, without

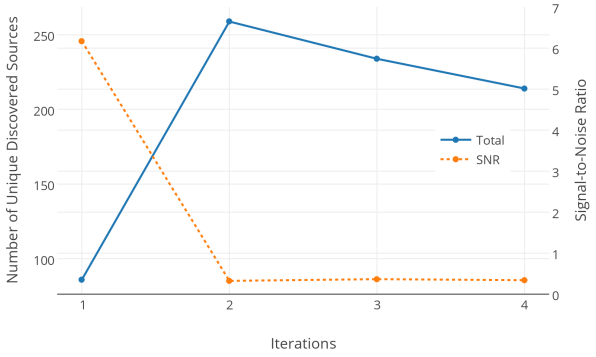
Iterations			
1	2	3	4
politics	abbott	abbott	abbott
australia	australia	government	tony
government	government	australia	australia
australian	news	tony	government
university	australian	news	news
party	minister	minister	minister
minister	party	party	party
abbott	politics	people	people
news	tony	pm	politics
media	people	politics	australian

(a) Normal search

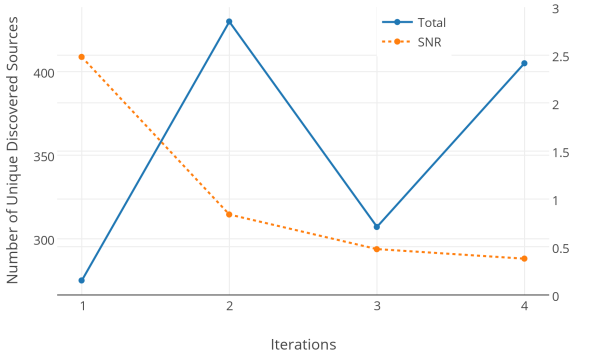
Iterations			
1	2	3	4
politics	australia	australia	abbott
australia	government	abbott	australia
australian	politics	government	rudd
government	party	rudd	labor
party	australian	party	government
news	abbott	minister	party
media	minister	labor	minister
world	labor	australian	news
minister	rudd	politics	australian
abbott	pm	news	election

(b) Broad search

Figure 8: Top 10 keywords used for searches



(a) Normal search



(b) Broad search

Figure 9: Signal-to-Noise Ratio of data sources

requiring new hardware deployments. The integration of disparate social and static media platforms also supports cross-cultural and cross-platform experimentation and analysis without requiring extensive user knowledge or experience.

This work also evaluates the use of a novel new data sourcing algorithm, designed to optimise the task of collecting relevant data for use in the framework. Two different approaches to sourcing are evaluated, with relevant data sources identified. A narrow search approach is found to discover relevant social media sources, while a broader search is more adept at discovering historical data sources. As the framework is designed to operate over both new and historical data, both approaches are useful.

As future work, improvements to the sourcing algorithm are recommended to extend its effectiveness to new and historical data sources simultaneously, providing the most relevant data sources possible for use in data analysis and applications.

References

Agichtein, E., Castillo, C., Donato, D., Gionis, A. & Mishne, G. (2008), Finding high-quality content in social media, *in* 'Proceedings of the 2008 International Conference on Web Search and Data Mining', ACM, pp. 183–194.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y. & Cayirci, E. (2002), 'Wireless sensor networks: a survey', *Computer networks* **38**(4), 393–422.

Bird, S., Klein, E. & Loper, E. (2009), *Natural language processing with Python*, O'Reilly Media, Inc.

Borsboom, B., Amstel, B. v. & Groeneveld, F. (2010), 'Please rob me'.
URL: *http://pleaserobme.com/*

Breslin, J., Bojars, U., Passant, A., Fernandez, S. & Decker, S. (2009), 'Sioc: Content exchange and semantic interoperability between social networks'.

Brickley, D. & Miller, L. (2000), 'The friend of a friend (FOAF) project'.
URL: *http://www.foaf-project.org/*

Burke, J. A., Estrin, D., Hansen, M., Parker, A., Ramanathan, N., Reddy, S. & Srivastava, M. B. (2006), 'Participatory sensing', *Center for Embedded Network Sensing* .

Cameron, M. A., Power, R., Robinson, B. & Yin, J. (2012), Emergency situation awareness from twitter for crisis management, *in* 'Proceedings of the 21st international conference companion on World Wide Web', ACM, pp. 695–698.

Campbell, A. T., Eisenman, S. B., Lane, N. D., Miluzzo, E. & Peterson, R. A. (2006), People-centric urban sensing, *in* 'Proceedings of the 2Nd Annual International Workshop on Wireless Internet', WICON '06, ACM, New York, NY, USA.

Chen, M., Gonzalez, S., Vasilakos, A., Cao, H. & Leung, V. C. (2011), 'Body area networks: A survey', *Mobile Networks and Applications* **16**(2), 171–193.

Ching, A., Murthy, R., Molkov, D., Vadali, R. & Yang, P. (2012), 'Under the hood: Scheduling MapReduce jobs more efficiently with corona'.

URL: *https://www.facebook.com/notes/facebook-engineering/under-the-hood-scheduling-mapreduce-jobs-more-efficiently-with-corona/10151142560538920*

DataSift (2010), 'DataSift'.
URL: *http://datasift.com/*

Doan, A., Domingos, P. & Halevy, A. Y. (2001), Reconciling schemas of disparate data sources: A machine-learning approach, *in* 'ACM Sigmod Record', Vol. 30, ACM, pp. 509–520.

Edison (2012), 'The social habit'.
URL: *http://socialhabit.com/secure/wp-content/uploads/2012/07/the-soCial-Habit-2012-by-edIson-research.pdf*

FormulaPR (2011), 'Social networking in summary'.
URL: *\url{http://www.formulapr.com/fuse/june2011/bitech.pdf}*

Ganti, R. K., Ye, F. & Lei, H. (2011), 'Mobile crowdsensing: Current state and future challenges', *Communications Magazine, IEEE* **49**(11), 32–39.

Gnip (2008), 'Gnip'.
URL: *http://gnip.com/*

Guo, B., Yu, Z., Zhang, D. & Zhou, X. (2014), 'From participatory sensing to mobile crowd sensing', *arXiv preprint arXiv:1401.3090* .

Hachem, S., Pathak, A. & Issarny, V. (2013), 'Service-oriented middleware for large-scale mobile participatory sensing', *Pervasive and Mobile Computing* .

Hardt, D. (2012), 'The OAuth 2.0 authorization framework'.

Hughes, D., Crowley, C., Daniels, W., Bachiller, R. & Joosen, W. (2014), User-rank: generic query optimization for participatory social applications, *in* 'System Sciences (HICSS), 2014 47th Hawaii International Conference on', IEEE, pp. 1874–1883.

Hughes, D., Greenwood, P., Blair, G., Coulson, G., Grace, P., Pappenberger, F., Smith, P. & Beven, K. (2008), 'An experiment with reflective middleware to support grid-based flood monitoring', *Concurrency and Computation: Practice and Experience* **20**(11), 1303–1316.

Hughes, D., Thoelen, K., Horré, W., Matthys, N., Cid, J. D., Michiels, S., Huygens, C. & Joosen, W. (2009), LooCI: a loosely-coupled component infrastructure for networked embedded systems, *in* 'Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia', ACM, pp. 195–203.

Kanhere, S. S. (2011), Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces, *in* 'Mobile Data Management (MDM), 2011 12th IEEE International Conference on', Vol. 2, IEEE, pp. 3–6.

Khoury, R., Dawborn, T., Gafurov, B., Pink, G., Tse, E., Tse, Q., Almi'Ani, K., Gaber, M., Röhm, U. & Scholz, B. (2010), Corona: energy-efficient multi-query processing in wireless sensor networks, *in* 'Database Systems for Advanced Applications', Springer, pp. 416–419.

Lane, N. D., Eisenman, S. B., Musolesi, M., Miluzzo, E. & Campbell, A. T. (2008), Urban sensing systems: opportunistic or participatory?, *in* 'Proceedings of the 9th workshop on Mobile computing systems and applications', ACM, pp. 11–16.

Li, J. & Cardie, C. (2013), 'Early stage influenza detection from twitter', *arXiv preprint arXiv:1309.7340* .

Madden, S. R., Franklin, M. J., Hellerstein, J. M. & Hong, W. (2005), 'TinyDB: an acquisitional query processing system for sensor networks', *ACM Transactions on database systems (TODS)* **30**(1), 122–173.

Maynard, D., Bontcheva, K. & Rout, D. (2012), 'Challenges in developing opinion mining tools for social media', *Proceedings of NLP* .

Min, H., Scheuermann, P. & Heo, J. (2013), 'A hybrid approach for improving the data quality of mobile phone sensing', *International Journal of Distributed Sensor Networks* **2013**.

Okada, H., Itoh, T., Suzuki, K. & Tsukamoto, K. (2009), Wireless sensor system for detection of avian influenza outbreak farms at an early stage, *in* 'Sensors, 2009 IEEE', IEEE, pp. 1374–1377.

Perry, F. (2014), 'Sneak peek: Google cloud dataflow, a cloud-native data processing service'.
URL: *http://googlecloudplatform.blogspot.com/2014/06/sneak-peek-google-cloud-dataflow-a-cloud-native-data-processing-service.html*

Robinson, B., Power, R. & Cameron, M. (2013), A sensitive twitter earthquake detector, *in* 'Proceedings of the 22nd international conference on World Wide Web companion', pp. 999–1002.

Sakaki, T., Okazaki, M. & Matsuo, Y. (2010), Earthquake shakes twitter users: real-time event detection by social sensors, *in* 'Proceedings of the 19th international conference on World wide web', pp. 851–860.

Sato, K. (2012), 'An inside look at google BigQuery, white paper', *Google Inc* .

Singapore, W. A. S. (2014), 'Social, digital & mobile in APAC'.
URL: *http://www.slideshare.net/wearesocialsg/social-digital-mobile-in-apac*

Taylor, D., Lewin, J. & Strutton, D. (2011), 'Friends, fans, and followers: Do ads work on social networks?', *Business Faculty Publications* .

Vasalou, A., Joinson, A. N. & Courvoisier, D. (2010), 'Cultural differences, experience with social networks and the nature of "true commitment" in facebook', *International Journal of Human-Computer Studies* **68**(10), 719–728.

Vassiliadis, P. (2009), 'A survey of extract–transform–load technology', *International Journal of Data Warehousing and Mining (IJDWM)* **5**(3), 1–27.

Wandhöfer, T., Taylor, S., Walland, P., Geana, R., Weichselbaum, R., Fernandez, M. & Sizov, S. (2012), 'Determining citizens' opinions about stories in the news media: analysing google, facebook and twitter', *eJournal of eDemocracy & Open Government (JeDEM)* **4**(2), 198–221.