SQL or NoSQL? Contrasting approaches to the storage, manipulation and analysis of spatio-temporal online social network data

Adrian Tear

Department of Geography, University of Portsmouth, United Kingdom
adrian.tear@port.ac.uk

Abstract. Researchers are now accessing millions of Online Social Network (OSN) interactions. These are available at no or low cost through Application Programming Interfaces (APIs) or data custodians including DataSift and GNIP. Records held in Extensible Markup Language (XML) or JavaScript Object Notation (JSON) are well structured but often inconveniently formatted for use in popular Relational Database Management Systems (RDBMS) or Geographic Information Systems (GIS) software. In contrast, emerging NoSQL (Not-only Structured Query Language) technologies are specially designed to 'ingest' unstructured data. Extract/Transform/Load (ETL) procedures for the storage and subsequent analysis of two OSN datasets in SQL/NoSQL databases are examined. The fixed data model of the relational approach may prove problematic when loading unpredictable document-based structures arising from extended periods of data collection. Although relational databases are far from obsolete the spatial analysis community seems likely to benefit from experimentation with new software explicitly designed for handling spatio-temporal Big Data.

Keywords: SQL, NoSQL, XML, JSON, tabular, document, databases

1 Introduction

Massive recent growth in Online Social Network (OSN) usage, along with low or no-cost data availability has prompted much new research, particularly in the social sciences. JISC [1, p3] note that 'Vast amounts of new information and data are generated everyday through economic, academic and social activities. This sea of data, predicted to increase at a rate of 40% p.a., has significant potential economic and societal value. Techniques such as text and data mining and analytics are required to exploit this potential.' The increasing use of 'geo-tagged' content on OSNs, now described by some as 'Geo Social Networks' [2–4], has resulted in growing interest [5, 6] and novel journals such as *Mobile Media & Communication* investigating the characteristics of this new 'locative media' [7]. Often derived from web sources – such as Twitter, Facebook or Flickr – these new forms of spatio-temporal data present particular computational challenges for researchers generally more familiar with the intersection of tabular datasets/database systems and desktop Geographic Information Systems (GIS). Web-

based OSN data are typically well structured, but are often inconveniently formatted for use in popular Relational Database Management System (RDBMS) or GIS mapping software. The Extensible Markup Language (XML) or JavaScript Object Notation (JSON) formats commonly used for data interchange store records in a leaf-node 'document' model [8–10]. Extract/Transform/Load (ETL) procedures required to parse and normalize arbitrarily defined XML or JSON document data into a relational model using Structured Query Language (SQL) will a) require significant skill on the part of the operator and, b) may increase the possibility for introduced error during the import stages. Furthermore, with very large datasets, complex or long-running ETL processes require significant computing resources or may fail, requiring time-consuming re-coding, on all but the best computer hardware. Recent advances in Not-only SQL (NoSQL) databases, and the contrasting approaches to the storage and manipulation of social media data as 'table' or 'document' are considered with reference to ongoing research into the spatio-temporal characteristics of OSN interactions recorded during recent (US 2012) and current (Scottish 2014) electoral events. NoSQL databases appear to offer an attractive alternative to traditional relational systems for the storage of fast-changing or potentially unpredictable document-based data structures arising from extended periods of web-based social media data collection

2 Research background

This research examines:

- ~1.7m records sampled from a 'Big Data' corpus of ~75m Twitter Tweets and Facebook Posts made in the run-up to the 2012 US Presidential Election.
- ~1.9m records consisting of Twitter Tweets made in the run-up to the Scottish Independence Referendum forthcoming in September 2014.

Following the US Presidential Election of 2008 Barack Obama was described as the 'first Internet President' [11–13] and in the run-up to the 2012 campaign 'it was clear that the war for the White House would be heavily fought online' [14, p1]. Rapid growth in OSNs and increasing online consumption of news and/or opinion before and during electoral campaigns [15–18] has prompted political parties to invest time, effort and money stimulating discussion and attempting to benefit from user interactions made over 'Web 2.0' social media channels [19]. Successful attempts have been made to raise funding, support levels and voter turnout using OSN sites such as Facebook and Twitter, particularly in the United States [20, 21]. Facebook, founded in 2004, now claims over 1.15bn active monthly users [22] whilst Twitter, founded in 2006, claims over 200m users posting over 400m Tweets per day [23]. During elections enormous numbers of messages regarding politics are spread amongst members of OSNs. The ability of candidates, such as Obama, to target communications with >29m Facebook Friends or >20m Twitter Followers is a remarkable innovation in the personalization of political communication. The potentially 'Orwellian' nature of these new forms of interaction [24] have been highlighted by one of Obama's campaign officials who has

reportedly stated that 'the information that is interesting to us is behavioral: we want to serve you with stuff that you are going to like' [25].

Although elections have provided a rich seam of study for researchers over many years the ability to analyze mass sentiment of electorates and 'political actors' [26] is relatively new-found. The growth of OSNs, geo-tagging and the public availability of 'Big Data' [27] have opened up new opportunities for research [1]. A'61-million-person experiment in social influence and political mobilization' to determine whether specific prompts on Facebook influence voter turnout through 'friend' recommendation and ease of use in finding 'local polling places' has been reported in *Nature* [28]. The online geographic spread of interest in events following a riot in the US and 'its manifestation within the geoweb' [29, p130] has been studied. Increased geo-codability of Twitter data has been reported [30] through mining of content in the text and metadata of over 1.5bn Tweets. OSN data have been used to report on local traffic conditions [31], location based services have been analyzed [32] and Volunteered Geographic Information (VGI) such as geo-coded Flickr photographs have been used to manage crisis events [33, 34]. There is growing interest in 'neogeography' [35, 36] and the study of individuals' real or online interaction with space.

Using an 'exploratory analysis' approach [37, p503] this study focuses on visualization and investigation of spatio-temporal social media usage during electoral campaigns. In line with published recommendations [29, p138] the research is intended to '[move] beyond the simple mapping and analysis of user-generated online content tagged to particular points on the earth's surface [to consider] the diversity of social and spatial processes, such as social networks and multi-scalar events, at work in the production, dissemination, and consumption of geoweb content.' In order to do so a range of analytical techniques have been (or will be) applied such that the choice of data storage/manipulation software may impact directly on the range of possible results. Two contrasting data storage technologies have been used. The relative merits of each for handling large spatio-temporal datasets are discussed in more detail below following a description of the data collection and extraction methodologies employed.

3 Data collection and extraction

Many of the major OSN operators provide Application Programming Interfaces (APIs) allowing web developers – or researchers – to query and collect feeds of publicly available social media 'interactions'; the Tweets, Facebook posts, Flickr photographs, URL links or other media uploaded or exchanged by users of OSN web sites. The code used to control each operator's API changes reasonably frequently [38] and may only provide access to a subset of the available data stream [39]. Access to full data streams, such as Twitter's 'Firehose', may be 'very hard to come by and potentially very expensive to realistically consume' [40]. As a result OSN operators have teamed with data aggregators who provide access to current and historic social media data. The two largest operators are DataSift (www.datasift.com) and GNIP (www.gnip.com), both of whom manage upstream API integration and provide a single-point-of-access to upwards of twenty individual social media data sources.

In this study DataSift – which operates a 'pay as you go' billing model – has been used to collect ~1.7m social media interactions in the run-up to the US Presidential Election of 6 November 2012 together with a further ~1.9m (and growing) interactions in the run-up to the 18 September 2014 vote on Scottish Independence. DataSift's Curated Stream Definition Language, CSDL [41], allows – at its simplest – social media messages to be filtered on the basis of content. The CSDL below, for example, will find all available social media interactions containing the phrase 'computer science'.

```
interaction.content contains_any "computer science"
```

More complex CSDL rules have been constructed to sample and extract records from Twitter and Facebook for US 2012 and Scottish 2014 elections using a range of text search terms (see http://tinyurl.com/appendix1-csdl) and controlling for explicit presence/absence of geographic coordinates, extent (country) and/or language. The interactions output from these CSDL definitions may be recorded and stored on Datasift's servers prior to download. The default data format used is JSON, 'a lightweight, text-based, language-independent data interchange format' that 'facilitates structured data interchange through a syntax of braces, brackets, colons, and commas' [10]. JSON can represent 'objects and arrays [which] nest [allowing] trees and other complex data structures [to] be represented' [10, p ii]. The format has been widely adopted by major OSN sites including Twitter [42] and Facebook [43] and may be illustrated by way of example, using a snippet of a Tweet in raw JSON format sent from the Twitter account of Presidential Candidate Barack Obama (Fig. 1).


```
{"interaction":{"author":{"ava-
tar":"http:\/\/a0.twimg.com\/profile_im-
ages\/2764236884\/90102995f6e328d7f90c43c8b337a0c7_nor-
mal.png","id":813286,"link":"http:\/\/twit-
ter.com\/BarackObama","name":"Barack
Obama","username":"BarackObama"},"content":"Happening
now: President Obama speaks in Ohio about the choice in
this election. RT so your friends can watch, too.
http:\/\/t.co\/d42qgdn8","created_at":"Mon, 05 Nov 2012
21:39:41 +0000
[…]
}
```


Fig. 1. Snippet of a JSON formatted Tweet from the account of Candidate Barack Obama created on 05/11/2012 (full file downloadable from http://tinyurl.com/obama-json; use a browser-based JSON viewer [44] to expand the data into a clear human-readable format)

Raw data are represented by a sequence of Unicode code points, certain characters (e.g. the solidus character or forward slash '/') are escaped and file encoding is UTF-8. The degree of nesting and the length of the arrays will vary from record to record. Around 150,000 records with explicit geographic coordinates resulted from sampling during

the run-up to the US 2012 Presidential Election. These records have another object (geo) nested within the Interaction object the data for which, in JSON, takes the form:

```
"geo":{"latitude":40.8183573,"longitude":-73.965401}
```

The ability of JSON to systematically describe arbitrarily defined data makes it both extremely powerful and potentially difficult to handle in traditional, tabular RDBMS and GIS software.

4 Data growth and value

In 1970 E.F.Codd [38, p377] stated that 'Future users of large data banks must be protected from having to know how the data is organized in the machine' describing a relational framework designed to provide 'the independence of application programs and terminal activities from growth in data types and changes in data representation'. Although well aware of the potential for 'natural growth in the types of stored information' Codd and the early designers of RDBMS software would probably not have anticipated the recent step-change in volumes of stored data and diversity of data types. A 2011 McKinsey research report [46] estimated (p103) that the amount of new data stored worldwide in 2010 amounted to >6,750 petabytes with >3,500 petabytes added in the United States and >2,000 petabytes added in Europe. The report suggested that the rate of data storage growth would exceed 20% per annum and, indeed, it is now commonplace to speculate on the 'whateverbyte problem' [47] of naming the unit that will follow petabytes (10^15 bytes), exabytes (10^18), zettabytes (10^21) and yottabytes (10^24) to describe the next scale of massive data storage. McKinsey's [46, p2] report 'finds that data can create significant value for the world economy, enhancing the productivity and competitiveness of companies and the public sector and creating substantial economic surplus for consumers.' Examples of savings or efficiencies in healthcare, transportation and government are highlighted as a result of 'the ability to generate, communicate, share, and access data [that] has been revolutionized by the increasing number of people, devices, and sensors that are now connected by digital networks.' Locational data, which currently makes up only a small proportion of worldwide data growth by volume, is seen by McKinsey [46, p38] as 'a nascent domain' of potentially high value '[cutting] across industry sectors from telecom to media to transportation [and featuring] a hotbed of innovation that could transform organizations and the lives of individuals.'

5 Data input, manipulation and storage

It has been stated [48, p v] that 'Big data are often differentiated from traditional large databases using the three Vs: volume, variety, and velocity.' The 3Vs have posed significant challenges to computer science in terms of storage (tabular or document-based, magnetic or solid state disk, vertical or horizontal scaling, private or public cloud), ma-

nipulation and analysis (SQL or NoSQL, XQuery, SPARQL, Hadoop and MapReduce). Many technical responses, including several of those mentioned above, have resulted from 'open-sourcing' of projects such as Google's Bigtable [49] 'distributed storage system', the foundation for Apache HBase [50], and related technologies such as the Hadoop Distributed File System (HDFS) [51] adopted and often improved upon by major technology companies or web site owners such as Facebook [52]. The pace of change in 'Big Data' technology is extremely rapid and since 'increasingly, location-aware datasets are of a size, variety, and update rate that exceeds the capability of spatial computing technologies' the difficulty of handling 'Spatial Big Data (SBD)' [53, p81] presents a particular use-case. The implications of moving away from a traditional RDBMS/GIS approach towards handling spatio-temporal Big Data are examined. Findings based on the manipulation of ~4m OSN interactions (in ~6GB of raw JSON data space) are presented below. The dataset is not big enough to be representative of some of the very major storage, manipulation or analysis problems occurring in significantly larger datasets but usefully highlights technical, workflow and scale issues of relevance to individual researchers or research teams.

5.1 Tabular data storage

Despite fairly early (1998) acknowledgement by the industry of the need 'to radically broaden [the database systems] research focus to attack the issues of capturing, storing, analyzing, and presenting the vast array of online data' [54, p74] tabular/relational database systems have, since Codd's [45] pioneering work, provided the foundation for much data storage, manipulation and analysis over the last thirty years or more. Tables of data 'normalized' into relationships have formed the basis of many existing researchers' professional training and operational experience. Commercial RDBMS software such as Oracle, DB2 or Microsoft SQL Server and more recent open-source products such as PostgreSQL or MySQL have been prevalent in both academia and the workplace for quite some time. GIS software from vendors including ESRI, MapInfo, Intergraph and others also typically relate tabular 'attribute' data to geometric features (points, lines or polygons) in order to provide processing capabilities which may query both spatial data and values stored in database rows and columns. More recently, relational systems have integrated 'spatial' data as a type in their own right, extending SQL capabilities in this direction and using Binary Large Objects (BLOBS) or similar to store otherwise 'unstructured' data including documents, images or video within the database management system.

RDBMSs are generally characterized by the need for a database schema defining relationships between tables. With 'clean-sheet' designs schemas may be created, typically in advance, using a range of Information Technology (IT) workflow practices such as 'Unified Modelling Language' (UML) [55] or 'PRojects IN Controlled Environments' (PRINCE or PRINCE2) [56]. These approaches have been designed to capture and document processes, flows and data usage following a period of skilled analysis. However, much of the web-based data available today forms an awkward fit with 'designed-in-advance' schemas or models based on understandings of current business or data consumption practices.

In this research three streams of social media data covering the same event (the 2012 US Presidential Election) were collected using the same data extraction engine (DataSift CSDL [41]) over a roughly two month period. Although the three resultant datasets were JSON-based the extraction engine allowed for download in Comma-Separated Values (CSV) format, converting the leaf/node document structure to a series of rows and commonly named columns. Consequently – and as a result of established working practice, preference and expediency – initial ETL processes read data from these three CSV files into the Microsoft SQL Server RDBMS. Fig. 2 shows the CSV field commonality (over 146 fields, effectively the number of unique JSON key/value pairs) across the three streams collected.

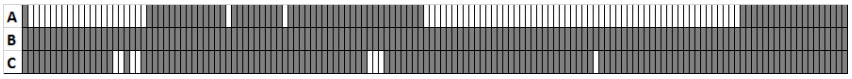

Fig. 2. CSV field commonality across three OSN data streams (A: US_2012_GEO, B: US_2012_NON_GEO and C: US_2012_NON_GEO_HISPANIC) collected using DataSift during the US 2012 Presidential Election

Only stream B's CSV file provided the superset of all fields. Crucially, however, stream A (US_2012_GEO) was the first to be defined in DataSift CSDL with OSN recording commencing on 4 September 2012. Had the number of fields in stream A (n=67) been used to define a fixed database schema for data storage at that time then neither streams B (n=146, commenced 6 September 2012) nor C (n=138, commenced 5 October 2012) would have fitted the data model.

As a result of this incongruity across data streams – a result of the ability of JSON to systematically describe arbitrary data, the transformation of this data from JSON to CSV and the possibility that upstream API changes during the period of data collection had altered the number of fields – a sustained effort was required to import the three US 2012 CSV files containing the data streams recorded and to bring them all into a common table-based relational model. Fig. 3 shows a high level representation of the several stages involved in this ETL process. Altogether 57 SQL scripts were written (over a period of around one month) to import, check, convert or re-tabulate data. The purpose of each script cannot be covered in depth but some of the key findings are worth highlighting:

- Initial imports using the 'Import Data Wizard' of an older version of SQL Server's Data Transformation Services (DTS) resulted in field truncation and data loss.
- Two import attempts failed to correctly handle UTF-8 encoded strings resulting in data loss both of international characters (e.g. Spanish diacritics) and emoticons etc.
- The transformation of JSON arrays to delimited strings in CSV fields (e.g. `[''var1'', ''var2'', ''var3'']`) required row-based normalization.
- Various post-processing `CAST` or `CONVERT` statements were required following data import e.g. to correctly handle long date formats.

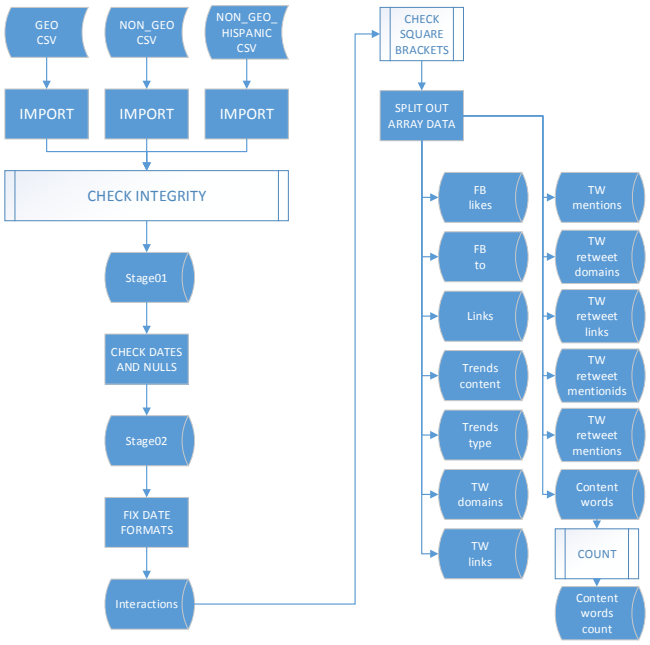

Fig. 3. Schematic representation of process stages and tables in the US_2012 SQL database

Problems with handling UTF-8 encoded text have subsequently been overcome using SQL Server (2012) Data Transformation Tools (SSDT) [57] and changed working practice [58]. However, the many import steps required, the effort involved in designing SQL statements to reformat data and the slow-running nature of some `UPDATE` queries on a long/wide table (all of which would have been magnified considerably with a much larger dataset) prompted the search for database software able to natively handle JSON formatted data.

5.2 Document data storage

As the amount of unstructured or semi-structured data generated by human, web or sensor-based activities has grown technologists have developed a raft of software products designed to store and interrogate 'documents'. This work is not new [59] but has come of age alongside the 'extraordinary information explosion [seen] over the last decade' [48, p v]. Around 150 different document store approaches are available [60] including those based around the Hadoop/MapReduce 'ecosystem' [61] or NoSQL database products such as MongoDB [62] or MarkLogic Server [63].

In this research MarkLogic Server was used. The software is comparatively well-established in its sector, having first been developed in 2001 to handle multi-terabyte XML document data storage and interrogation using XQuery [64]. Although a commercial product MarkLogic is, like Microsoft SQL Server, available with a free and fully functional developer license. The Server software runs on several flavours of

Unix/Linux, on 64-bit Windows and as an Amazon Machine Image in Amazon's Elastic Cloud Compute (EC2) architecture [65]. Fig. 4 shows a high level representation of the comparatively few stages (c.f. Fig. 3) involved in the MarkLogic ETL (and application deployment) process.

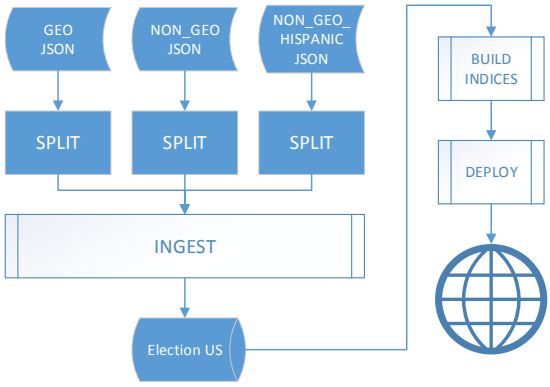

Fig. 4. Schematic representation of process stages and tables in the US_2012 NoSQL database

In this test:

- Both US 2012 (three) and Scottish 2014 (one) data streams were re-exported from DataSift servers, this time using the JSON file format.
- Examination of these files showed UTF-8 encoding with one JSON record on each row the row terminator being the Line Feed (LF) character.
- For convenience of 'ingestion' each of the four JSON files (~4m rows in all) were split into *n* individual JSON files using the Linux command `split -dl 1 --additional-suffix=.json [SOURCE.JSON] record` to create *n* output files (e.g. `recordc9999028134.json`) in the file directory system.
- A new blank database was created in MarkLogic Information Studio and a Data Flow task defined to ingest the files in these directories, using the optional switch to transform JSON to XML.
- A number of Element Range indices and a GeoSpatial Element Pair index were built to facilitate 'faceted search' on certain fields and mapping of Latitude/Longitude geo-referenced records.
- A web-based MarkLogic visualization application was built and deployed using the Application Builder tool within Information Studio.

Subsequent work has determined that the `SPLIT` stage shown in Fig. 4 is unnecessary. Using the MarkLogic Content Pump (MLCP) software it is possible to ingest line feed-delimited files either from the file system or compressed (.ZIP or .GZ) archives [66]. As MarkLogic Server runs an embedded web server, JSON data may also be ingested in real time over Hypertext Transfer Protocol (HTTP) using RESTful APIs.

6 Data analysis and outputs

Both approaches to the input, manipulation and storage of spatio-temporal OSN data proved successful. However, the operator experience and workflows proved very different. The tabular/SQL approach used CSV files and required multiple import attempts, checks and extensive post-processing to create a workable 'clean' database using familiar tools. The document/NoSQL approach provided straightforward ingestion of JSON formatted data into a schema-agnostic document database offering 'out of the box' web-based application development using less familiar tools.

Thus far most analysis has been performed using RDBMS SQL queries with spatial analysis performed in the MapInfo desktop GIS. This represents a 'traditional' approach to analysis and visualization that will be familiar to many practitioners; using different software packages and moving data between packages best suited or most commonly used to count, plot, overlay, graph (see http://tinyurl.com/US2012-presentation) or visualize records (see http://tinyurl.com/US2012-animation). Whilst these activities reflect the exploratory research design [37] and help to reveal geographic patterns of spatio-temporal OSN usage there are several shortcomings:

- The software 'stack' is not tightly integrated.
- The approach is unlikely to scale well with very large datasets.
- There is little possibility to interactively step forward or back through time.
- The words in OSN text are easy to count but difficult to interpret.

As the research progresses the ability to analyze text is expected to be a key requirement. The US 2012 dataset consists of ~1.7m time-stamped OSN messages, many of which may be geo-referenced directly or indirectly through text matching. The corpus contains >30m words with ~1.4m distinct words. The Scottish 2014 dataset consists of ~1.9m OSN messages and rising and will inevitably comprise a broadly similarly sized corpus. It is clear that 'the huge amount of free-form unstructured text in the blogosphere, its increasing rate of production, and its shrinking window of relevance, present serious challenges to the […] analyst who seeks to take public opinion into account' [67]. Technology comparisons in other disciplines, such as the largely text-based world of clinical data storage [68], have suggested that 'while NoSQL and XML technologies are relatively new compared to the conventional relational database, both of them demonstrate potential to become a key database technology.' Relational databases offer fully-developed facilities such as `SELECT…GROUP BY…` or `SELECT…ORDER BY…` to query, aggregate, count or order data but most were never designed for the detailed analysis of free-form text. In contrast NoSQL databases have, in many cases, been explicitly designed to handle free text or web-based mark-up languages. Preliminary investigations have been made using the MarkLogic Server database in order to evaluate the applicability of a NoSQL/XML approach to the storage, spatio-temporal and textual analysis of opinion-rich social media data. Relevant features include:

- **Ease of 'ingestion'** – Experience has shown that loading large numbers of OSN documents in JSON format into MarkLogic is straightforward. For those interested

in real-time analysis the software may also ingest records (e.g. from Twitter) through a RESTful API direct to the database.

- **Alerting** – Alerts may be used to highlight text by re-writing XML content or may fire a cascade of other events to find or enrich entities or build semantic relationships. Alerting of this type is widely used in the intelligence community, one of the early adopters of NoSQL technology.
- **Entity extraction and enrichment** – Can be used to find, e.g., all matches of 'Chicago' re-writing content to add XML tags such as `<placename lat=41.88 lon=-87.62>Chicago</placename>` which can be used in subsequent textual or geospatial analysis.
- **Semantic enrichment** – Individual records may be linked to Resource Description Framework (RDF) triples enabling subject-predicate-object analysis with SPARQL. Triple store relationships describing, e.g., Town to State geographies, should enable query and analysis of OSN records at multiple geographic scales.
- **Text handling** – MarkLogic has well-developed facilities for text handling and search; custom dictionaries, custom thesauri, word stemming, near word matching and so on. Features such as thesaurus expansion can, e.g., be used in sentiment analysis to match synonyms of 'good' or 'bad' within *n* words of a candidate's name.
- **Tight database/web integration** – MarkLogic is both a database and web server, capable of clustering/load-balancing at data and application levels and providing public or restricted access to content and functionality through scripts written in XQuery without the need for separate database and webserver/middleware layers.
- **Horizontal scaling and multi-tiered storage** – With the dataset sizes under consideration it is unlikely that MarkLogic's horizontal scaling (using, e.g., multiple Amazon Machine Images in Amazon EC2) or multi-tiered storage (e.g. archival on Amazon S3, recent on magnetic disk, latest on solid state disk) will be used. The ability to 'spin up' multiple instances to parallelize analysis may prove more useful.

Although the features above have been identified largely for their potential to improve upon RDBMS' capabilities for text-based analysis of spatio-temporal OSN messages it is somewhat unlikely that any one database, or database technology, will provide a panacea for all data-related storage or analysis requirements. Users must frame the questions and write the code to extract maximum analytical benefit from the underlying technology; the choice of technology simply sets the bounds of what is possible – or, more accurately, what is possible most easily or straightforwardly based on operator knowledge and experience of the system. The entity enrichment and semantic possibilities offered by MarkLogic Server appear particularly useful but even here some have warned [67] that 'while the structural elements of Web 3.0 lend themselves quite well to graph-theoretical identification of communities or communicating blogs […] they have done relatively little to identify the content of blog posts and comments by topic so as to permit classification and clustering.' Self-organizing maps (SOMs), support vector machines (SVMs) and other emerging machine learning technologies attempt to automate text classification through statistical means and are most often modular add-ons to mathematical or statistical analysis software. Elsewhere open source

text analysis 'ecosystems'– such as Sheffield University's open source General Architecture for Text Engineering (GATE) – offer extremely advanced features to analyze massive amounts of text stored in files, in RDBMSs such as Oracle or PostgreSQL or in the GATE cloud service [69]. These systems must also be examined to determine which approach offers the most successful and efficient means of deriving meaning from millions of words of spatio-temporally referenced text. This 'plumbing' or 'knitting together a patchwork of different components into integrated workflows' [70] is one of the key challenges of Big Data mining and will, today, almost certainly involve the use of multiple technologies.

At this stage in the research programme it is not possible to state whether a SQL or NoSQL database represents the best (or only) fit with the requirement to quantify, map, visualize and explain differences between 'geographic' and 'non-geographic' users of OSN sites during electoral periods. Both technologies have pros and cons; many features in tabular/SQL databases are extremely well-understood and fully developed whereas document/NoSQL approaches are currently less widely adopted and hence somewhat less well-understood. Neither of the database technologies enable the full range of sophisticated geographic analyses possible in a GIS, but neither does a GIS provide capabilities for potentially massive data storage. Either technology may integrate with other desktop or server software (e.g. MatLab, GATE, Tableau, R) or with web-based Software as a Service (SaaS) offerings such as OpenCalais semantic enrichment or GATEcloud.net text analysis. Some of these SaaS products may even obviate the need for a large-scale user data store altogether by ingesting files themselves and returning metadata, reports and analysis as the output. Therefore, as is so often the case in Information Technology, competing products offer differing approaches to problem-solving which – depending upon a range of factors including input format, dataset size, availability/cost, preference and expediency – may present equally valid, or at least viable, solutions for given use cases. Unless or until a high performance, large scale and potentially all-encompassing 'CyberGIS' is developed [71] it seems likely that those managing, analyzing and visualizing text heavy spatio-temporal OSN data will continue to integrate a number of products or technologies to fulfil their individual operational or research objectives.

7 Summary

Contrasting approaches to the storage, manipulation and analysis of spatio-temporal Online Social Network data have been described with reference to ongoing research into the use of social media during electoral events. The two OSN datasets discussed in this study are sized well within the capabilities of the SQL, NoSQL and GIS software products used. However, even at this scale, it is apparent that a data model fixed at design time may prove problematic when handling fast-changing or potentially unpredictable document-based data structures arising from extended periods of social media data collection. If the datasets were 100x, 1,000x or 10,000x larger or the data were fast-changing or streaming in real-time, the various challenges already identified would be magnified considerably. 'Ease of ingestion' would, in this case, quite probably tip

the balance in favour of the NoSQL approach, even if lack of familiarity with the technology required time for learning in order to carry out effective downstream analysis. Capturing high volume, highly variable and high velocity data will at least allow later analysis whereas a broken parsing or import routine to a pre-defined RDBMS schema will simply result in data loss.

More work is required to benchmark contrasting SQL and NoSQL approaches to Big Data ETL, storage, analysis and total cost of ownership [48]. It also seems likely that operator and workflow experiences will require just as much research. Long-term familiarity with SQL/RDBMS software, allied to the cost and complexity of setting up clustered cloud environments running the latest 'bleeding-edge' software, may limit the uptake of new NoSQL technologies outside all but the most highly technical research or computer science departments. Nonetheless, there is much the spatial analysis community can learn, even using virtual machines running on commodity laptop of desktop hardware, through experimentation with new Big Data technologies explicitly designed to handle extremely large, often web-based, spatio-temporal datasets.

References

1. JISC: The Value and Benefit of Text Mining to UK Further and Higher Education. Digital Infrastructure. (2012).
2. Campbell, S.W., Kwak, N.: Political Involvement in "Mobilized" Society: The Interactive Relationships Among Mobile Communication, Network Characteristics, and Political Participation. J. Commun. 61, 1005–1024 (2011).
3. Lee, C.-H.: Mining spatio-temporal information on microblogging streams using a density-based online clustering method. Expert Syst. Appl. 39, 9623–9641 (2012).
4. Bahir, E., Peled, A.: Identifying and Tracking Major Events Using Geo-Social Networks. Soc. Sci. Comput. Rev. 31, 458–470 (2013).
5. Licoppe, C.: Merging mobile communication studies and urban research: Mobile locative media,"onscreen encounters" and the reshaping of the interaction order in public places. Mob. Media Commun. 1, 122–128 (2013).
6. Humphreys, L.: Mobile social media: Future challenges and opportunities. Mob. Media Commun. 1, 20–25 (2013).
7. Wilken, R.: Locative media: From specialized preoccupation to mainstream fascination. Converg. Int. J. Res. into New Media Technol. 18, 243–247 (2012).
8. W3C: Extensible Markup Language (XML), http://www.w3.org/XML/.
9. www.json.org: JSON, http://www.json.org/.
10. ECMA International: ECMA-404 The JSON Data Interchange Format. , Geneva (2013).
11. Pew Research Center's Project for Excellence in Journalism: McCain vs. Obama on the Web: A Study of the Presidential Candidate Web Sites, http://www.journalism.org/node/12772.
12. Greengard, S.: The first internet president. Commun. ACM. 52, 16–18 (2009).

13. Levenshus, A.: Online Relationship Management in a Presidential Campaign: A Case Study of the Obama Campaign's Management of Its Internet-Integrated Grassroots Effort. J. Public Relations Res. 22, 313–335 (2010).
14. Towner, T.L.: All Political Participation Is Socially Networked? New Media and the 2012 Election. Soc. Sci. Comput. Rev. 1–15 (2013).
15. Polat, R.K.: The Internet and Political Participation: Exploring the Explanatory Links. Eur. J. Commun. 20, 435–459 (2005).
16. Mutz, D.C., Young, L.: Communication and Public Opinion: Plus Ca Change? Public Opin. Q. 75, 1018–1044 (2011).
17. Hong, S.: Online news on Twitter: Newspapers' social media adoption and their online readership. Inf. Econ. Policy. 24, 69–74 (2012).
18. Kim, Y.: The contribution of social network sites to exposure to political difference: The relationships among SNSs, online political messaging, and exposure to cross-cutting perspectives. Comput. Human Behav. 27, 971–977 (2011).
19. Nooralahzadeh, F., Arunachalam, V., Chiru, C.: 2012 Presidential Elections on Twitter -- An Analysis of How the US and French Election were Reflected in Tweets. 2013 19th Int. Conf. Control Syst. Comput. Sci. 240–246 (2013).
20. Campbell, H.: Barack Obama and Twenty-First Century Politics : A Revolutionary Moment in the USA. Pluto Press, London, GBR (2010).
21. Takaragawa, S., Carty, V.: The 2008 US Presidential Election and New Digital Technologies: Political Campaigns as Social Movements and the Significance of Collective Identity. Tamara J. Crit. Organ. Inq. 10, 73–89 (2012).
22. Facebook: Key Facts - Facebook Newsroom, http://newsroom.fb.com/content/default.aspx?NewsAreaId=22.
23. Tsukayama, H.: Twitter turns 7: Users send over 400 million tweets per day, http://articles.washingtonpost.com/2013-03-21/business/37889387_1_tweets-jack-dorsey-twitter, (2013).
24. Chamley, C., Scaglione, A., Li, L.: Models for the Diffusion of Beliefs in Social Networks: An Overview. IEEE Signal Process. Mag. 30, 16–29 (2013).
25. McGregor, R.: Obama campaign sharpens tech edge, http://www.ft.com/cms/s/0/b2e7043c-2284-11e1-923d-00144feabdc0.html, (2011).
26. Lees-Marshment, J., Lilleker, D.G.: Knowledge sharing and lesson learning: consultants' perspectives on the international sharing of political marketing strategy. Contemp. Polit. 18, 343–354 (2012).
27. Boyd, D., Crawford, K.: Critical Questions for Big Data. Information, Commun. Soc. 15, 662–679 (2012).
28. Bond, R.M., Fariss, C.J., Jones, J.J., Kramer, A.D.I., Marlow, C., Settle, J.E., Fowler, J.H.: A 61-million-person experiment in social influence and political mobilization. Nature. 489, 295–298 (2012).
29. Crampton, J.W., Graham, M., Poorthuis, A., Shelton, T., Wilson, M.W., Zook, M.: Beyond the geotag: situating "big data" and leveraging the potential of the geoweb. Cartogr. Geogr. Inf. Sci. 40, 130–139 (2013).
30. Leetaru, K., Wang, S., Cao, G., Padmanabhan, A., Shook, E.: Mapping the global Twitter heartbeat: The geography of Twitter. First Monday. 18, (2013).

31. Kosala, R., Adi, E.: Harvesting Real Time Traffic Information from Twitter. Procedia Eng. 50, 1–11 (2012).
32. Wilson, M.W.: Location-based services, conspicuous mobility, and the location-aware future. Geoforum. 43, 1266–1275 (2012).
33. Spinsanti, L., Ostermann, F.: Automated geographic context analysis for volunteered information. Appl. Geogr. 43, 36–44 (2013).
34. Goodchild, M.F., Glennon, J.A.: Crowdsourcing geographic information for disaster response: a research frontier. Int. J. Digit. Earth. 3, 231–241 (2010).
35. Warf, B., Sui, D.: From GIS to neogeography: ontological implications and theories of truth. Ann. GIS. 16, 197–209 (2010).
36. Batty, M., Hudson-Smith, A., Milton, R., Crooks, A.: Map mashups, Web 2.0 and the GIS revolution. Ann. GIS. 16, 1–13 (2010).
37. Andrienko, N., Andrienko, G., Gatalsky, P.: Exploratory spatio-temporal visualization: an analytical review. J. Vis. Lang. Comput. 14, 503–541 (2003).
38. Stieglitz, S., Kaufhold, C.: Automatic Full Text Analysis in Public Social Media – Adoption of a Software Prototype to Investigate Political Communication. Procedia Comput. Sci. 5, 776–781 (2011).
39. Morstatter, F., Pfeffer, J., Liu, H., Carley, K.: Is the sample good enough? comparing data from twitter's streaming api with twitter's firehose. Proc. ICWSM. (2013).
40. Twitter: How do I get firehose access? | Twitter Developers, https://dev.twitter.com/discussions/2752.
41. DataSift: Language Guide | DataSift Developers, http://dev.datasift.com/csdl.
42. Twitter: Overview: Version 1.1 of the Twitter API | Twitter Developers, https://dev.twitter.com/docs/api/1.1/overview.
43. Facebook: JSON with Unity, https://developers.facebook.com/docs/unity/reference/current/Json/.
44. Firefox: JSONView :: Add-ons for Firefox, https://addons.mozilla.org/en-US/firefox/addon/jsonview/.
45. Codd, E.F.: A Relational Model of Data for Large Shared Data Banks. Commun. ACM. 13, 377–387 (1970).
46. Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C., Hung Byers, A.: Big data: The next frontier for innovation, competition, and productivity. (2011).
47. Foley, J.: OracleVoice: Extreme Big Data: Beyond Zettabytes And Yottabytes - Forbes, http://www.forbes.com/sites/oracle/2013/10/09/extreme-big-data-beyond-zettabytes-and-yottabytes/.
48. Rabl, T., Poess, M., Baru, C., Jacobsen, H.: Specifying Big Data Benchmarks. Springer Berlin Heidelberg, Berlin, Heidelberg (2014).
49. Chang, F.A.Y., Dean, J., Ghemawat, S., Hsieh, W.C., Wallach, D.A., Burrows, M., Chandra, T., Fikes, A., Gruber, R.E.: Bigtable: A Distributed Storage System for Structured Data. ACM Trans. Comput. Syst. 26, 4:2–4:26 (2008).
50. Apache: HBase - Apache HBase™ Home, http://hbase.apache.org/.
51. Apache: Welcome to Apache™ Hadoop®!, http://hadoop.apache.org/.

52. Borthakur, D., Rash, S., Schmidt, R., Aiyer, A., Gray, J., Sarma, J. Sen, Muthukkaruppan, K., Spiegelberg, N., Kuang, H., Ranganathan, K., Molkov, D., Menon, A.: Apache hadoop goes realtime at Facebook. Proc. 2011 Int. Conf. Manag. data - SIGMOD '11. 1071 (2011).
53. Shekhar, S., Evans, M.R., Gunturi, V., Yang, K., Cugler, D.C.: Benchmarking Spatial Big Data. Specifying Big Data Benchmarks. pp. 81–93. Springer (2014).
54. Bernstein, P., Brodie, M., Ceri, S., DeWitt, D., Franklin, M., Garcia-Molina, H., Gray, J., Held, J., Hellerstein, J., Jagadish, H. V, others: The Asilomar report on database research. ACM Sigmod Rec. 27, 74–80 (1998).
55. D'Souza, D.F., Wills, A.C.: Objects, components, and frameworks with UML: the catalysis approach. Addison-Wesley Reading (1998).
56. Axelos: About PRINCE2® | PRINCE2®, http://www.prince-officialsite.com/AboutPRINCE2/AboutPRINCE2.aspx.
57. Microsoft: Microsoft Download Center, http://www.microsoft.com/en-us/download/details.aspx?id=36843.
58. Murray, S.: Import UTF-8 Unicode Special Characters with SQL Server Integration Services, http://www.mssqltips.com/sqlservertip/3119/import-utf8-unicode-special-characters-with-sql-server-integration-services/.
59. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Commun. ACM. 35, 61–70 (1992).
60. Edlich, S.: NOSQL Databases, http://nosql-database.org/.
61. Cutting, D.: The Apache Hadoop Ecosystem, http://assets.en.oreilly.com/1/event/75/The Apache Hadoop Ecosystem Presentation.pdf.
62. MongoDB: MongoDB, http://www.mongodb.org/.
63. MarkLogic: Enterprise NoSQL Database | MarkLogic, http://www.marklogic.com/.
64. Walmsley, P.: XQuery. O'Reilly (2009).
65. MarkLogic: MarkLogic 7 — MarkLogic Developer Community, http://developer.marklogic.com/products.
66. MarkLogic: Using MarkLogic Content Pump (Loading Content Into MarkLogic Server) — MarkLogic 7 Product Documentation, http://docs.marklogic.com/guide/ingestion/content-pump.
67. Till, B.C., Longo, J., Dobell, a. R., Driessen, P.F.: Self-organizing maps for latent semantic analysis of free-form text in support of public policy analysis. Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 4, 71–86 (2014).
68. Lee, K.K.-Y., Tang, W.-C., Choi, K.-S.: Alternatives to relational database: comparison of NoSQL and XML approaches for clinical data storage. Comput. Methods Programs Biomed. 110, 99–109 (2013).
69. Cunningham, H., Tablan, V., Roberts, A., Bontcheva, K.: Getting more out of biomedical documents with GATE's full lifecycle open source text analytics. PLoS Comput. Biol. 9, e1002854 (2013).
70. Lin, J., Ryaboy, D.: Scaling big data mining infrastructure: the twitter experience. ACM SIGKDD Explor. Newsl. 14, 6–19 (2013).
71. Wang, S.: CyberGIS: blueprint for integrated and scalable geospatial software ecosystems. Int. J. Geogr. Inf. Sci. 27, 2119–2121 (2013).