Mind The Gap: Newsfeed Visualisation with Metro Maps

Damask Talary-Brown

Bachelor of Science in Computer Science with Honours
The University of Bath
2017

Signed: 

Mind The Gap: Newsfeed Visualisation with Metro Maps

Submitted by: Damask Talary-Brown

COPYRIGHT

Declaration

This dissertation is submitted to the University of Bath in accordance with the requirements of the degree of Bachelor of Science in the Department of Computer Science. No portion of the work in this dissertation has been submitted in support of an application for any other degree or qualification of this or any other university or institution of learning. Except where specifically acknowledged, it is the work of the author.

Signed: 

Abstract

Contents

List of Figures

List of Algorithms

Introduction

"The Press, Watson, is a most valuable institution, if you only know how to use it."
— Sherlock Holmes, *The Adventure of the Six Napoleons*
Sir Arthur Conan Doyle

Aims

The aim of this project is the development of a tool which generates interactive metro maps of data from RSS feeds, with individual news articles transformed into stations and common themes transformed into *metro lines*. My goal is to reduce the information overload experienced by news consumers, through the provision of contextual links and topic background within the visualisation of the feeds.

The resultant system is a news feed aggregator with graphically structured output, and to the best of my knowledge is first of its kind.

Outline

The structure of this dissertation is as follows:

Chapter 1 provides an overview of the background literature upon which this project relies, including a detailed exploration of alternative visualisation formats.

Chapter 2 describes the scoping of the system, the informal requirements gathering process undertaken, and the rationale behind the significant design decisions.

Chapter 3 describes the algorithms and techniques chosen to transform the data from feed to visualisation and discusses how they were implemented in the system.

Chapter 4 provides a set of example results generated by the system and discusses the effect of altering various parameters on the data.

Chapter 5 describes the process by which the system was evaluated.

Chapter 6 summarises both the contributions and limitations of this work, and provides a discussion on future research directions.

Chapter 1

Literature and Technology Review

1.1 Why don't we Understand the News?

The day the result of the 2016 United Kingdom EU Membership Referendum was announced, the @GoogleTrends Twitter account reported a 250% increase in searches for "What happens if we leave the EU?" Much like the case of David Leonhardt's 2008 article in the New York Times in which began, "Raise your hand if you don't quite understand this whole financial crisis," national news commentary had focused on little else in the preceding months.

Some months after Leonhardt's article was published, Journalism Professor Jay Rosen voiced his agreement with its premise in a blog post on the failure of journalism during the financial crisis; "there are certain very important stories – and the mortgage crisis is a good example – where until I grasp the whole I am unable to make sense of any part."[Rosen, 2008]

It has become apparent that prolific coverage alone is not enough to engage and support the public in understanding the complexities of current events. Historically, news media has been limited in the volume of content it can produce by physical constraints such as printing costs, but the rise of the internet as a platform to deliver it has lead to an explosion of content, both through existing media channels and through competing social media websites and blogs.

The term *ambient news* was coined by Hargreaves et al. [2002] to describe the ubiquity of news in the current information landscape. Others have commented in a more critical light; describing the proliferation of competing news media as "as pervasive–and in some ways as invasive–as advertising." [Nordenson, 2008, p.2]

In 2008, The Associated Press conducted an extensive field study into the news consumption habits of young adults. Among their key findings were three points which acutely summarise the news overload problem;

- **"Consumers are experiencing news fatigue."**

  The study found participants were debilitated and overwhelmed, and that their levels of dissatisfaction lead to a decrease in the effort they put into news acquisition. This is consistent with multiple other studies [Holton and Chyi, 2012, Purcell et al., 2010, Fischer and Stevens, 1991] which found participants across every demographic were overwhelmed by the amount of news content available to them and agreed that it prevented them exploring news on less familiar topics.

- **"Story resolution is key."**

  Participants' consistent enjoyment of sports and entertainment news was due in part to the formulaic storytelling which characterises these types of journalism, with clear chronology to provide contextual back story. The feeling of enjoyment gained from reading procedural stories directly contrasts with what the same participants experienced reading World news, where they struggled to find resolution to stories which were unfolding at the time.

- **"Consumers want depth but aren't getting it"**

  It was observed that participants, in their efforts to discover *below-the-fold* content (defined in the context of the AP's model, 2008, p.37) from particular headlines, often found themselves reading the same summary-level content from different news sources. It was recommended that news providers support this by "designing innovative formats and creating easier pathways to deep content." [Associated Press and Context-Based Research Group, 2008, p.49]

Initially, the third point seems to be a direct contradiction to the first; we are overwhelmed by the volume of news we are exposed to, but we also crave more detail from the news we do consume. However, it brings to light the issue of information *quality* as a requirement of news consumers.

Journalism, and therefore its quality, can be viewed along a spectrum between two models; a model for the communication of facts, and a model for entertainment and storytelling. From the three points above, it is apparent that quality at both ends of the spectrum is being sought, since the desire for quality below-the-fold content is covered by the first model, and the desire for quality story resolution by the second.

1.1.1 Information Overload

News fatigue is a domain-specific type of information overload, a phenomenon formally defined as "when the information processing demands on time to perform interactions and internal calculations exceed the supply or capacity of time available for such processing" [Schick et al., 1990, p.206]. Information overload is a multifaceted problem which can be modelled as a combination of three contributing factors (Figure 1.1).

These factors correspond directly to the three points previously identified from the Associated Press and Context-Based Research Group study. High information quantity leads to news fatigue, information format determines the level of possible story resolution, and information quality determines how much depth a reader can gain from the news they consume. The authors did not find a single solution which could address all three factors, but they did identify information quantity as the most significant contributor to overload.

Bergamaschi et al. [2010] further decompose information quantity into spatial and temporal dimensions in the specific context of news articles. Spatial quantity refers to articles which are near-identical in terms of facts presented being published by different media outlets, and temporal quantity refers to articles on a single topic being published in quick succession over a short period of time.

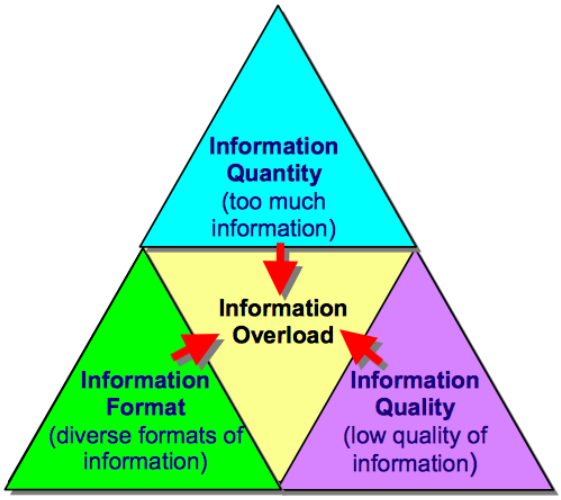


Figure 1.1: Dimensions of information overload, as defined by Ho and Tang [2001].

Intuitively, the terms spatial and temporal quality seem illogically named, as a set of articles with high spatial quantity would cover a smaller area of information space and vice versa. High spatial quantity will therefore be referred to as *redundancy*, and high temporal quantity as *fragmentation*, since a sudden burst of articles published on the same topic suggests a currently unfolding story being told in parts.

To adequately determine the contributory factors relating to news fatigue, all four dimensions of information overload should be considered, and will therefore be explored in more detail in the following sections.

Information Quality

In the context of factual data rather than news specifically, Strong et al. [1997] defined information quality in terms of four components; intrinsic quality, accessibility quality, contextual quality and representational quality. If news can be rationalised to its core function as an interpretation of facts and other raw data such as images, then this same framework can be experimentally applied to news journalism in order to determine which factors could influence its quality.

Intrinsic quality is a measure of the accuracy, objectivity, believability and reputation of data. In the context of news, the first three factors would typically be true for all major news sources, and the reputation would be dependent on whether the article originated from a trusted source or not. Accessibility quality is less relevant to online news media, as it is concerned with data access and security. Contextual quality is the most relevant category in respect to news, concerning timeliness, amount of data, and value-added. In the news domain, this would mean an article's quality is dependent on its performance against a background of other articles; whether or not it contributes anything recent or previously unknown. Finally, representational quality is concerned with ease of understanding and interpretability, which are easily translatable concepts.

The implications of applying the Information Quality Framework [Strong et al., 1997] to news articles are that quality may be influenced by the reputation of the source, the timeliness of publication, value-added by the article (i.e. content which couldn't be derived from other sources) and the ease of understanding of the content.

Information Format

The domain of news articles is a more specific information space than that of documents in general, and by nature most news articles share some common formatting and structural elements such as headlines, timestamps, and relevant images. As a result of this, it is unlikely that any two articles from popular news providers would be diverse enough in content format to overshadow the information quantity problem.

Fragmentation (Temporal Quantity)

The rise of social media sites such as Twitter delivering news to consumers has lead to a high degree of news fragmentation, due to the constraints of the microblogging service's 140-character limit. 24-hour television news paved the way for new formats of real-time content delivery, and the ever-expanding network of online social media channels stepped up to deliver.

It logically follows from the fragmented nature of real-time news journalism that temporal quality suffers; stories are published and updated intermittently over short periods of time, meaning there is more content for the consumer to piece together in order to understand a story. The fragmentation is somewhat mitigated by Twitter's use of hashtags to denote a Tweet's topics. Hashtags help readers form a coherent and picture of unfolding events from the incremental contributions of thousands of participating users [Bruns et al., 2012].

Phuvipadawat and Murata [2010] developed a methodology to collect and group Tweets on breaking news topics, using hashtags for topic identification or *story-finding*, and grouping similar messages together to form a single news story. Their algorithm for similarity is a function of the TF-IDF [Salton and Buckley, 1988] of the two messages and the number of named entities they have in common.

Redundancy (Spatial Quantity)

It is in the nature of news that newsworthy stories get repeated across multiple sources. When consumers read news on a particular from more than once source, it is likely that they will read variations on the same facts in multiple articles.

Attempts such as [Barzilay et al., 1999] have been made to synthesise summaries of collections of similar online documents, a practice here termed *information fusion*, with news articles from different sources being given as a specific use-case. However, the process of extracting common sentences between documents was in order to reformulate them into a single summary, rather than to determine the level of similarity between the documents.

A more relevant approach was presented in [Pera and Ng, 2008], where the *title* and *description* attributes of elements in RSS feeds were used as content descriptors to mitigate the overhead

of processing entire documents for phrases. The content descriptors are then used to compute phrase n-grams as a measure of similarity between any two documents. The similarities in this case were used to remove subsumed articles and cluster non-redundant similar ones, in order to streamline feed content for readers.

It should be noted that there is an overlap between the notion of spatial quantity and one of the four influencing factors in information quality; contextual quality. If a feed contains two articles which state the same number of identical facts, they therefore contribute to information overload on both the qualitative and quantitative fronts.

Viewing the dimensions of overload from a news domain perspective, it is clear that (consistent with the findings of Ho and Tang [2001]) information quantity is the most relevant contributing factor in respect to fatigue, along factors influencing contextual quality such as value-added and timeliness. Any proposed solutions to the news overload problem should therefore address these factors first.

1.1.2 Supporting Sensemaking

Sensemaking is the basis for forming contextual knowledge; the process by which we incorporate new information into our existing cognitive frameworks, and how we go from reading something to understanding it [Pentina and Tarafdar, 2014]. In broader terms, Weick et al. describe sensemaking as "[being] about the question: What does an event mean? In the context of everyday life, when people confront something unintelligible and ask, 'What's the story here?'" [Weick et al., 2005, p.85]

This definition relates directly to the news overload problem because one component of sensemaking is the contextual story resolution The Associated Press and Context-Based Research Group [2008] study identified news consumers are craving. It has also been observed that often readers are not interested in specific articles on a subject, and only the thematic content of the topic they belong to [Husin et al., 2014]. How then, do readers make sense of a collection of articles surrounding a particular topic?

When presented with a large document collection, Russell et al. [2006] found all of their subjects began by clustering the contents into groups which formed a heuristic representation or mental model, used to provide an overview. However, current information infrastructure has been criticised for not supporting the cross-correlation between connected news articles [Rennison, 1994].

Writing for the Columbia Journalism review in 2008, Nordenson outlined a suggestion for the new roles of journalism in the information era; "By linking stories to one another and to background information and analysis, news organizations help news consumers find their way through a flood of information that without such mediation could be overwhelming and nearly meaningless"[Nordenson, 2008, p.10].

Similarly, Pentina and Tarafdar [2014] make the recommendation in the context of contemporary media consumption that news providers should adapt to an environment of news overload by adding facilities enabling readers to categorise, sort and search news collections. Additional findings of this study suggested that the contextual background provided by having more detailed coverage aids the sensemaking process, as it helps users form links between

new information and their existing frameworks, but this presents an interesting conflict with the goal of reducing information overload when considering large collections of documents.

It is apparent that many recommendations have been made from within the field of journalism that at the point of delivery, news content should incorporate contextual links between related articles. This is important both from a sensemaking perspective to emphasise connections, and from an information overload perspective to help users find meaning in an inundated news landscape.

The news overload problem can now be reformulated with scope and detail: How can we display a collection of related news articles in such a way that users are not overwhelmed by unstructured content and are free to explore the underlying contextual pathways?

A simple starting point comes from a familiar idiom; a picture is worth a thousand words.

1.2 An Overview of Information Visualisation

Of course, a picture is not always worth a thousand words, particularly when the picture is unstructured and complex in its own right. However, a recognised and effective technique for bridging the gap between a set of data and a user's mental model and subsequent comprension of the data is information visualisation, or InfoVis. [Yi et al., 2008, Havre et al., 2002]. This section provides a brief overview of a formative InfoVis taxonomy and uses the taxonomy to categorise appropriate visual models for newsfeed visualisation.

In his seminal paper on information visualisation, Shneiderman proposed a taxonomy for visualisations comprising seven data type abstractions, and seven tasks which are components of the visual information seeking mantra; "Overview first, zoom and filter, then details-on-demand." [Shneiderman, 1996, p.1]

Shneiderman's type abstractions are as follows:

1-dimensional	Linear data, where each datum is a string of characters.
2-dimensional	Planar data, e.g. layout diagrams, or clustered document collections.
3-dimensional	Physical objects or models of real-world entities, e.g. computer aided designs or medical imaging data.
Multi-dimensional	Any data where items with n attributes can be represented in n-dimensional space, e.g. relational databases, or feature vectors for classification.
Temporal	Data following a timeline, which is a subset of 1-dimensional data but was deemed important enough to warrant its own category. E.g. Project management data, or multimedia content timelines.
Tree	Hierarchical data where each datum has exactly one parent and zero or many children, e.g. document or directory structures.
Network	Related data, where each datum can have an arbitrary number of links to other data.

Because of the non-spatial nature of textual data, any visualisation of such data must involve some form of content abstraction and translation into a physical space [Wise et al., 1995]. These translations can result in data of arbitrary dimensionality, so a text corpus could fall into the 1-dimensional or multi-dimensional categories. News articles as a specific subset of textual data have certain metadata associated with them including dates, meaning they also fit the temporal type abstraction. In addition to this, if contextual links are considered part of the structure of the data, articles can be modelled as a network of connected nodes.

This ambiguity is not a failure of the taxonomy; Shneiderman stresses that composite categories are equally valid. However, the implications of this are that the most appropriate visualisation for a news corpus may itself be a composite of visualisations for any of its type abstractions, leaving an unfeasible number of possibilities to consider.

To reduce the scope of suitable visualisations, we return to the original problem of information overload. This time however, the aim is to minimise the overload from interpreting the model, rather than the overload from interpreting the data. Complex visualisations which require a considerable amount of effort to understand in their own right should be avoided where possible when reducing overload is the goal.

1.2.1 InfoVis for Sensemaking

In addition to the insight that visualisation may be able to provide, there is evidence that visual metaphors better support the learning process and are more easily remembered than isomorphic text representations alone [O'donnell et al., 2002, Yen et al., 2012].

Yi et al. [2008] identified four overlapping InfoVis processes which describe how insight can be gained after sensemaking; *Provide Overview*, *Adjust*, *Detect Pattern*, and *Match Mental Model*. These four processes can be roughly mapped to Shneiderman's high-level tasks, which are as follows:

Overview Gain a birds-eye view of the entire collection, with the option to change the scale of the view by zooming or using fisheye magnification techniques.

Zoom Gain a more detailed view of a portion of data or single datum while preserving the original sense of context.

Filter Nondestructively remove uninteresting data points or groups from the view.

Details-on-Demand Gain additional insight into one or more data points by selecting particular elements.

Relate View and explore relationships between elements.

History If necessary, undo previous actions to return to the a view of the data.

Extract Export selected data, preserving the format, for uses such as "sending by email, printing, graphing, or insertion into a statistical or presentation package" [Shneiderman, 1996, p.5].

The *Provide Overview* process allows a reader to recognise what they know and what they don't know from the information they are processing. The corresponding task in [Shneiderman, 1996] is *Overview*.

Adjust allows them to change the level of abstraction or field of selection of that information. This corresponds to *Zoom* and *Filter* in Shneiderman's task model.

The *Detect Pattern* procedure is where structure and trends are found (whether expected or otherwise). Coupled with *Match Mental Model*, where the links are formed between the new data and the users' existing cognitive frameworks, this corresponds to *Relate*.

At this point, Shneiderman's taxonomy diverges from the processes of [Yi et al., 2008], as Yi et al. are concerned with the cognition enabled by visualisation, whereas Shneiderman additionally considers other use cases for visualisations, such as querying and sharing.

From these two models, it is apparent that certain views and functions are crucial for tools which use visualisation to support the sensemaking process; an high-level overview visualisation which emphasises links between data, the ability to adjust scope to show more or less detail, and the ability to filter information of specific interest within the dataset.

1.2.2 Visual Metaphors

Eppler defines visual metaphors as "a graphic structure that uses the shape and elements of a familiar natural or man-made artefact or of an easily recognizable activity or story to organize content meaningfully and use the associations with the metaphor to convey additional meaning about the content." [Eppler, 2006, p.203] This definition highlights the main advantage to using visual metaphors; users are intuitively familiar with how they present and structure information.

The use of preexisting visual metaphors–specifically those with which a large number of people will already be familiar–has been shown to support readers' comprehension, as it requires both significant time and effort for a reader to interpret visual metaphors which are new to them [Ziemkiewicz and Kosara, 2009].

In a previous paper, Eppler also describes six advantages of visual metaphors specifically for the transfer of knowledge: "(1) to motivate people; (2) to present new perspectives; (3) to increase remembrance; (4) to support the process of learning; (5) to focus the attention of the viewer and (6) to structure and coordinate communication." [Burkhard, 2004, p.2, citing [Eppler, 2004]]. These are all desirable attributes, but motivation and support in the learning process are particularly relevant in addressing news fatigue.

Examples of visual metaphors commonly used to represent collections of data include calendars, bookshelves, timelines, maps and other schematics. Metaphors based on physical objects do not have to be visually skeuomorphic to be effective, but to avoid misinterpretation, there should be a match between the underlying structure of the metaphor and the underlying structure of the data. Two common and highly structured visual metaphors will be explored in the following sections in the context of potential for news representation; timelines and schematic maps.

Timeline Visualisations

Chronological ordering is an important characteristic of news articles and should be preserved in any visualisation of news data as it provides a natural ordering [Binh Tran, 2013]. Perhaps the simplest visual metaphor for a collection of dated documents is the timeline.

Nguyen et al. [2014*b*] explore the role of timelines in the sensemaking process, emphasising that the interactions supported by such visualisations should be as intuitive as possible in order to not disrupt users' trains of thought, and should be tightly coupled with other elements of the sensemaking process so temporal connections are not viewed solely in isolation.

Criticisms of previous timeline visualisations made by the authors are that linear layouts are often too simple for the data they represent, and that a lack of automatic layout generation results in additional manual work for the user.

The colouring technique used to distinguish sets of related events within a single timeline is a flexible extension to [Nguyen et al., 2014*a*], where the authors coloured events belonging to multiple sets with a gradient composed of the colours of both sets. However, the gradient approach presented in [Nguyen et al., 2014*b*] does not scale to events belonging to more than two sets, since the colour grouping restricts the number of possible intersections of each set. From a news storyline perspective, this would place an upper limit of two on the number of possible topics a story could belong to, which is a low constraint for all but the highest level topics.

Singh et al. [2015] designed a prototype for generating annotated timelines based on the Wikipedia entries long-running news stories. The use of Wikipedia rather than newsfeeds meant their document retrieval model was heavily dependent on Wikipedia's structure, but it also afforded a huge wealth of contextual information that made such detailed annotations possible. Not all stories are long running however, so while this would be useful as a retrospective tool it would be impossible to generate timelines in the same way for news articles which did not already belong to a long-running chain of events.

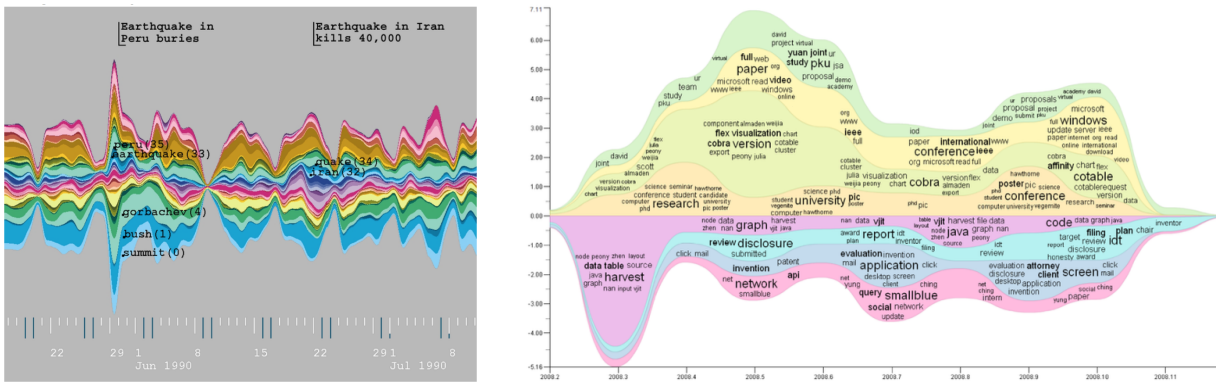


Figure 1.2: Similar visualisations from ThemeRiver [Havre et al., 2002] (left) and TIARA [Liu, Zhou, Pan, Qian, Cai and Lian, 2009] (right).

Both ESTHETE [Goyal et al., 2013] and nReader [Wang et al., 2006] present timeline-centric views for collections of news articles based on underlying graphs of relationships between the articles. However, in both cases, the graph structure was not part of the final visualisation, so

connections between entities were displayed in purely textual forms. In contrast, ThemeRiver [Havre et al., 2002] introduces a novel view on topic frequency along the time axis to show thematic change over time within a collection of documents, similar to a smoothed histogram. This view, while useful for large document collections which span weeks or months, would be less suited to displaying emerging news trends over shorter periods.

TIARA [Liu, Zhou, Pan, Qian, Cai and Lian, 2009] whose authors cite ThemeRiver as an influencing design (see figure 1.2), displays a similar shaped graphical output but performs more detailed textual analysis, and displays related keywords in the output. Both visualisations support simple zooming and panning, but suffer from the same limitations on visualising topic connectivity as [Nguyen et al., 2014*a*].

Taking into consideration both the critiques of oversimplification identified in [Nguyen et al., 2014*b*] and the physical limitations highlighted in [Nguyen et al., 2014*a*, Havre et al., 2002], it is clear that timelines may not be the most appropriate visual metaphor for news visualisation, especially not for highly connected events and topics. However, for more linear storylines which span fewer categories or topics, a visualisation such as [Nguyen et al., 2014*b*] could be used, for example as part of Shneiderman's *Zoom and Filter* task where the dataset is pruned.

Topological and Schematic Map Visualisations

The use of cartographic representations for abstract objects and the relationships between them is such a common and natural metaphor that it is often unnoticed in digital contexts. Maps take advantage of humans' natural ability to perceive and organise in a spatial context, and [the fact that we live in a spatial world] "leads naturally to metaphors that provide cues for orientation and navigation" [Old, 2002, p.2].

Topological maps[1] are a specific class of map which abstract away detail so that only significant features within desired subsets of the mapped dataset remain; from a news overload perspective, this is a highly desirable property. These maps are often used to visualise networks and are represented as schematics, where elements on the map are transformed into abstract visual representations for ease of understanding. Today, the most recognisable examples of topological maps are transit maps. Divorced from the strictness of geographical accuracy and scale, emphasis is instead placed on the usability of the map for planning and the understanding of relative positioning which the maps enable [Hochmair, 2009].

Figures 1.3 and 1.4 show the geographically accurate 1930 London Underground map designed by Frederick Stingemore, and the iconic 1933 redesign by Henry Beck, which was based on the concept of an electrical circuit diagram [Transport For London, 2014]. While Beck's simplification of the structure of the map was seen as radical and even controversial at the time, the hallmarks of his design are now recognisable not just in other schematic maps, but in posters, infographics and diagrams across various domains.

In terms of usability, simple timeline visualisations can provide the chronological ordering which is intrinsic within collections of news articles. However, schematic maps may be able to represent complex relationships between topics in a structure which is linear but has an additional dimension, to prevent linearity becoming a design or usability constraint.

[1] Not to be confused with *topographic* maps, which represent relief and other geographic features of physical regions at a large scale and in fine detail.

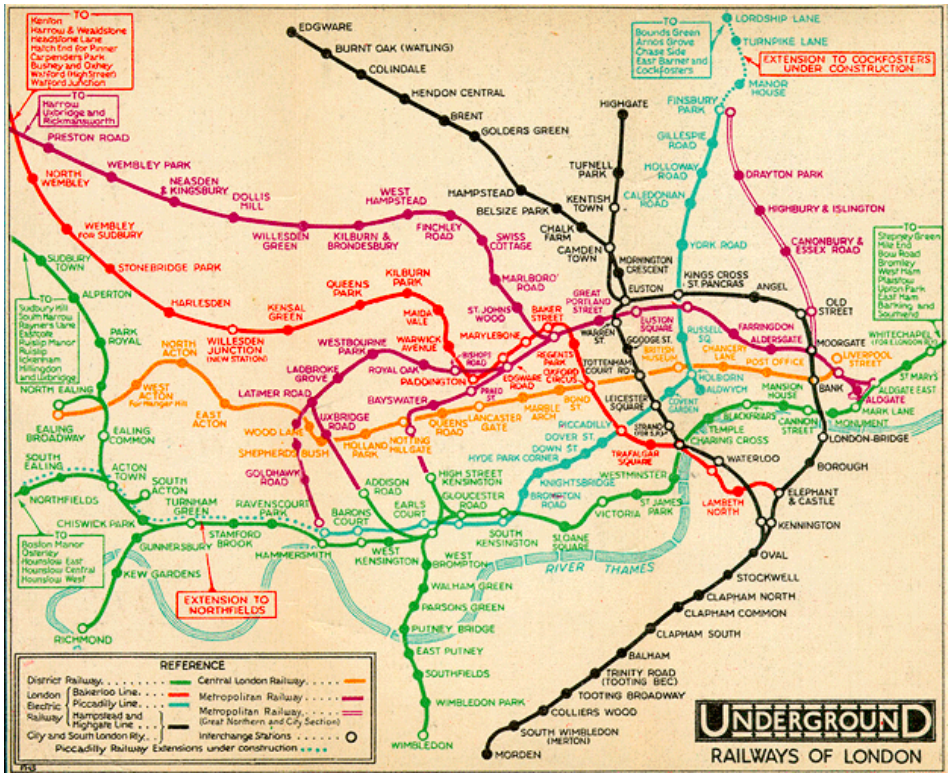


Figure 1.3: 1930 London Underground map, designed by F. Stingemore [British Broadcasting Corporation, 2013].

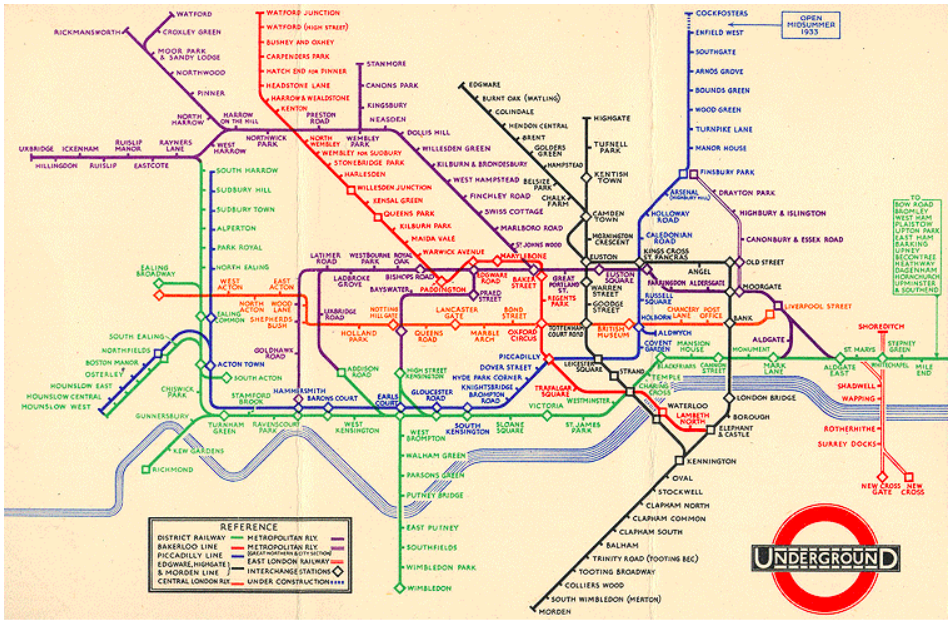


Figure 1.4: 1933 London Underground Map, designed by H. Beck [British Broadcasting Corporation, 2013].

1.3 Metro Maps for Information Cartography

Significant work in the area of information cartography has been undertaken by Shahaf et al. [Shahaf and Guestrin, 2010, Shahaf et al., 2012*b*,*a*, 2013], in the domains of both news and science through the visualisation of article and journal data on metro maps.

Shahaf et al. [2012*b*] chose the metaphor as a base for their visualisation to address the fact that previous timeline-based summarisation systems could only represent simple linear stories; "In contrast, complex stories display a very non-linear structure: stories split into branches, side stories, dead ends, and intertwining narratives." [Shahaf et al., 2013, p.1]

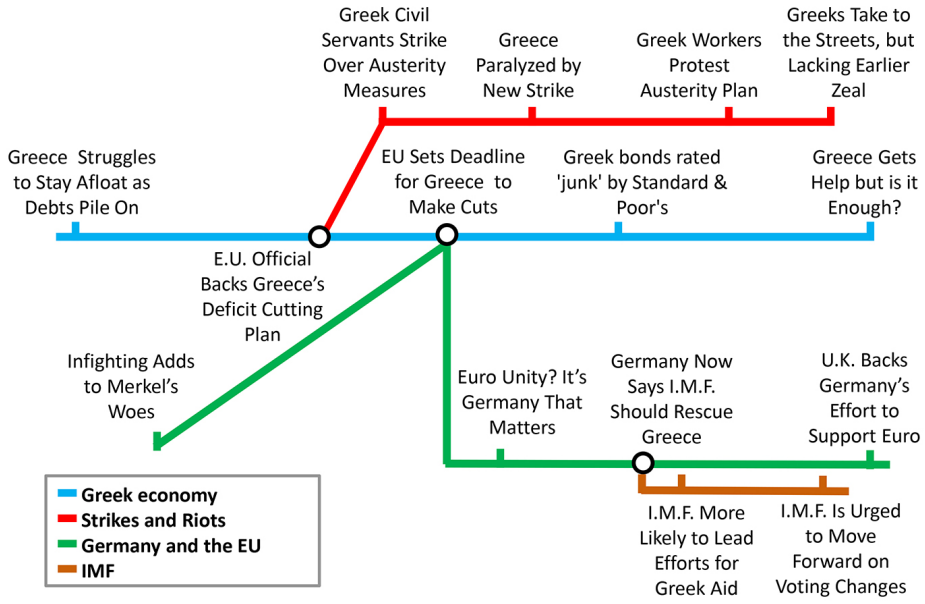


Figure 1.5: A metro map [Shahaf et al., 2012*b*] covering the Greek Debt Crisis.

Even in a scientific context, this was not the first time the abstract visualisation potential of the metro map had been noted; "The usefulness of the metro map as a metaphor is somewhat limited to simple examples by the time required to manually produce these maps. As such they are generally only useful for applications that do not change frequently. This limitation could be removed by quality methods for the automatic drawing of metro maps from abstract data." [Stott, 2008, p.54]

In this section, the formalisation of the metro map metaphor, its associated characteristics, and its limitations will be discussed.

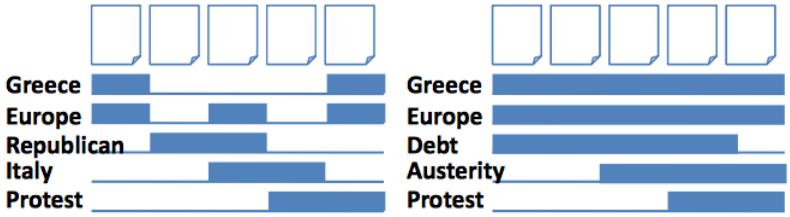
Definition 1. *Metro Map [Shahaf et al., 2012*b*]: A metro map* $\mathcal{M}$ *is a pair* (G, Π)*, where* $G = (V, E)$ *is a directed graph and* Π *is a set of paths, or metro lines in* G*. Each* $e \in E$ *must belong to at least one metro line.*

A previously published method [Shahaf and Guestrin, 2010] for linking together chains of articles was discussed, and an objective function was created to formalise the characteristics of a 'good' metro map. The function defined was a composite based on three important

characteristics, all of which are broadly applicable to the visualisation of any similar corpora; coherence, coverage, and connectivity.

1.3.1 Coherence

Let $\mathcal{D}$ be a set of articles, and $\mathcal{W}$ be a set of words or phrases, such that each article is a subset of $\mathcal{W}$. A *coherent* chain of articles through $\mathcal{D}$ is one where transitions between documents are smoothed by common overlapping keywords from $\mathcal{W}$, creating a better narrative flow [Shahaf and Guestrin, 2010] as depicted in Figure 1.6.



A bar corresponds to the presence of a word in the article above it.
The titles of the articles which made up the two chains were as follows:

Chain A (left)	Chain B (right)
Europe weighs possibility of debt default in Greece	Europe weighs possibility of debt default in Greece
Why Republicans don't fear a debt default	Europe commits to action on Greek debt
Italy; The Pope's leaning toward Republican ideas	Europe union moves towards a bailout of Greece
Italian-American groups protest 'Sopranos'	Greece set to release austerity plan
Greek workers protest austerity plan	Greek workers protest austerity plan

Figure 1.6: An incoherent chain with jittery transitions between topics (Chain A, left) alongside a more coherent chain of articles (Chain B, right). [Shahaf et al., 2012*b*]

Coherence, intuitively, seems to be closely linked to idea of story resolution detailed in Section 1.1.1. This presents a question which could later be explored further; does forming coherent chains of articles provide the story resolution that participants in the Associated Press study were so desperately seeking from current events journalism?

1.3.2 Coverage

As in the previous section, let $\mathcal{D}$ be a set of articles, and $\mathcal{W}$ be a set of words or phrases of which the articles are composed. The coverage function for a word in a given document $d_i \in \mathcal{D}$ specified in Equation 1.1 can be quantified using any measure of how well d_i covers w, for example TF-IDF$(w, d_i, \mathcal{D})$ (See Equation 1.6) [Shahaf et al., 2012*b*].

$$cover_{d_i}(w) : \mathcal{W} \rightarrow [0, 1] \tag{1.1}$$

Extending the notion of coverage to maps–which can be abstracted to sets of documents–introduces the idea of *diversity*. If a map already contains documents which for a sufficient coverage for some word w, then there is nothing to be gained by adding another document to $\mathcal{D}$ which has high coverage of w alone. This relates back to the principles of spatial and information quality discussed in Section 1.1.1, especially the importance of value-added by

every individual document in a collection. In this case, maps which cover a maximal number of $w \in \mathcal{W}$ should be preferential. A simple additive definition for map coverage such as Equation 1.2 [Shahaf et al., 2012*b*] would not reward this kind of diversity;

$$cover_{\mathcal{M}}(w) = \sum_{d_i \in docs(\mathcal{M})} cover_{d_i}(w) \tag{1.2}$$

Therefore, an alternative definition for map coverage was chosen, which will not increase significantly if another document which covers an already covered feature is added to $\mathcal{D}$ (Equation 1.3 [Shahaf et al., 2012*b*]).

$$cover_{\mathcal{M}}(w) = 1 - \prod_{d_i \in docs(\mathcal{M})} (1 - cover_{d_i}(w)) \tag{1.3}$$

Finally, the definition of map coverage is extended to the coverage of the corpus $\mathcal{D}$, rather than just single features. If each feature is weighted according to frequency, then for each $w \in \mathcal{W}$ we have some λ_w. The coverage of a corpus $\mathcal{D}$ by a metro map $\mathcal{M}$ can then be defined as in Equation 1.4 [Shahaf et al., 2012*b*].

$$Cover(\mathcal{M}, \mathcal{D}) = \sum_{w \in \mathcal{W}} \lambda_w cover_{\mathcal{M}}(w) \tag{1.4}$$

1.3.3 Connectivity

The final property is the most simply defined; the connectivity of a metro map is the number of paths in Π which intersect [Shahaf et al., 2012*b*].

$$Connectivity(\mathcal{M}) = \sum_{i<j} \mathbb{1}(p_i \cap p_j \neq \emptyset) \tag{1.5}$$

1.3.4 Limitations of [Shahaf et al., 2012*b*, 2013]

Corpus

Perhaps the biggest limitation of the system developed in [Shahaf et al., 2012*b*, 2013] is the nature of the corpus $\mathcal{D}$; it is a fixed dataset, meaning users can only query it for certain past events with no way of specifying a different corpus themselves.

From a historical reference perspective the output generated based on certain queries is interesting when compared to the output an expert would select as important to the narrative, but it is not possible to use the system as a replacement to a newsfeed aggregator.

Graph Layout Aesthetic Principles

From a usability perspective, a second limitation of the work of Shahaf et al. is the lack of focus given to the desirable aesthetic properties of transit maps.

A formative empirical study [Purchase et al., 1997] on how graph layout affects usability identified a collection of five measurable aesthetic principles from previous research, which aid human understanding when reading graphs:

- Minimise the number of line bends [Tamassia, 1987];
- Minimise the number of edge crossings [Ferrari and Mezzalira, 1970];
- Preserve any underlying symmetry in the structure of the graph [Lipton et al., 1985];
- Draw orthogonally where possible [Tamassia, 1987];
- Maximise the minimum angle between incident edges for each node [Garg and Tamassia, 1994].

These criteria, when numerically calculated and weighted relative to one another, allow the aesthetic quality of any graph to be evaluated. This is particularly useful for graphs generated by automatic layout algorithms, which aren't as much a product of human design intuition as their manually designed counterparts.

The origins of the principles above predate much of the early InfoVis research because several of them [Tamassia, 1987, Ferrari and Mezzalira, 1970] were formalised as guidelines for the design of electronic circuits.

In a later study, Purchase evaluated information-finding task performance with a set of graphs which varied the above principles, to establish which the most and least significant aesthetics were for graph usability. The results of the study were that maximising incident edge angles and orthogonality did not lead to better performance, preserving symmetry and minimising line bends were somewhat important, and minimising edge crossings was the most significant influencing factor for performance [Purchase, 1997].

Metro maps, however–particularly those representing non-physical data such as in the previous section–do not share all the structural principles of the wider set of directed graphs. Consequentially, Stott et al. developed a set of criteria for metro maps specifically, including criteria for the labelling of stations. The criteria are as follows [Stott et al., 2011]:

- **Angular Resolution Criterion:** Maximise the angle between incident edges at each node. This criterion is also the fifth principle in [Purchase et al., 1997].
- **Edge Length Criterion:** All edges on the map should be approximately equal.
- **Balanced Edge Length Criterion:** The length of edges incident to a given node should be approximately equal.
- **Edge Crossings Criterion:** Crossings should be minimised. This criterion is also the second principle in [Purchase et al., 1997], also identified as the most important.
- **Line Straightness Criterion:** Edges on the same metro line should be collinear, i.e. they should form a 180° line through every station that the line passes through. This criterion relates closely to the first principle in [Purchase et al., 1997].
- **Octilinearity Criterion:** Edges should be drawn at multiples of 45°. This criterion is an extension to the fourth principle in [Purchase et al., 1997], as orthogonality is here analogous to rectilinearity.

Evaluating Figure 1.5 according to the list above, we observe that despite only containing 16 stations, the map unnecessarily violates five of Stott et al.'s six criteria–all except edge crossings.

Therefore, to improve the usability of the metro maps drawn in [Shahaf and Guestrin, 2010, Shahaf et al., 2012*b*,*a*, 2013], it would be advisable to attempt to satisfy more of the aesthetic principles for graphs and metro maps described in [Purchase et al., 1997, Stott et al., 2011].

1.4 Towards Newsfeed Visualisation

The fact that news articles form a fairly narrow class of document is an advantage from a visualisation design perspective, due to the common elements they share. Articles published by commercial news producers typically contain:

- A headline;
- A description, or *subhead*;
- A publish date;
- One or more categories to which the article belongs.

These attributes are useful for visualisation, since creating a spatial representation from text requires documents to be represented as vectors in high-dimensional feature space [Wise et al., 1995], and the presence of existing attributes makes articles more inherently comparable than their unstructured contents would be.

There is also a well-known existing standard for publishing links to articles with their metadata for use by other applications; RSS.

1.4.1 Content Retrieval

The de-facto web format for feed publishing is RSS (Rich Site Summary, or Really Simple Syndication.) The rise of the internet as a news platform has lead to many readers finding the most efficient method of reading news articles is to subscribe to various topic-specific newsfeeds and read what is automatically collated by their computers [Wang et al., 2006].

Although RSS–which is a subset of XML–is standardised[2], the practice of feed categorisation is not, meaning the granularity of topics which can be subscribed to is dependent on the publisher. This issue was addressed by Liu, Han, Noro and Tokuda [2009], with the design of a system which could essentially split or join existing RSS feeds to synthesise new ones based on user-specified keywords and queries.

Despite its shortcomings, RSS remains the most universal option for accessing feed content from a wide variety of news producers [O'Shea and Levene, 2011].

[2] `http://cyber.harvard.edu/rss/rss.html`

1.4.2 Keyword Extraction

Extracting relevant keywords from documents is not a new domain of research. Various methods have been presented, the most well-known being the intuitively logical TF-IDF (term frequency, inverse document frequency) [Salton and Buckley, 1988] which ranks the significance of a term t in a document d which belongs to a corpus C as follows:

$$\text{TF-IDF}(t) = \frac{Occurrences(t,d)}{WordCount(d)} \times log_e\left(\frac{|C|}{|\{c \in C \mid t \in c\}|}\right) \tag{1.6}$$

TF-IDF will extract the most unique keywords from a document within a corpus, because it penalises words which are common to many documents. However, in the context of a corpus of news articles, this uniqueness can lead to significant topic keywords being ignored because they appear with such frequency.

Bun and Ishizuka [2002] found that for news archive keyword extraction, a better alternative to TF-IDF is TF-PDF (term frequency, proportional document frequency) as it is not biased against frequently repeated keywords.

Using TF-PDF, articles are modelled as belonging to one of a finite number of sources or *channels* within a corpus. The weighting of a term from an article within a channel is in this case linearly proportional to its frequency in the channel and exponentially proportional to the number documents in the channel where it occurs. A term's total weighting is the sum of its weightings across all channels, as can be seen in Equation 1.7 [Bun and Ishizuka, 2002], where:

- D = The number of channels in the corpus;
- K_c = The total number of terms in channel c;
- F_{tc} = Frequency of term t in channel c;
- n_{tc} = The number of articles in channel c where term t occurs;
- N_c = The total number of articles in channel c.

$$\text{TF-PDF}(t, D, K) = \sum_{c=1}^{c=D} \frac{F_{tc}}{\sqrt{\sum_{k=1}^{k=K_c} {F_{kc}}^2}} \times \exp\left(\frac{n_{tc}}{Nc}\right) \tag{1.7}$$

The reliance of both TF-IDF and TF-PDF on a fixed background corpus results in a need to recompute the function for every document if any are added to or removed from the collection. This is impractical for large collections, and even in the case of large fixed collections it does not scale well, which has resulted in the development of other methods.

An approach derived from energy levels in quantum systems was proposed in Carpena et al. [2009], where keywords were extracted based on their spatial distributions within a single text. The theory behind the approach is that typically, keywords occurrences are distributed in significant frequency clusters throughout a document, whereas non-relevant words are distributed with uniform frequency (see Figure 1.7).

This technique allows relevant keywords to be distinguished from non-relevant common words with similar total frequencies without the use of a background corpus for comparison.

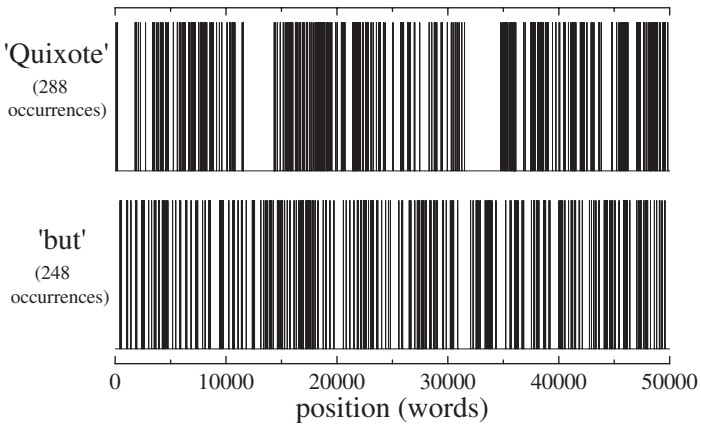


Figure 1.7: Frequency spectra for 'Quixote' and 'but' in the first 50,000 words of *Don Quixote*. [Carpena et al., 2009]

Several important observations have been made regarding keyword extraction for news articles specifically. Firstly, that important phrases in text are likely to be references to people, places and other named entities [Teitler et al., 2008]. Libraries such as the Stanford Named Entity Recognizer (NER) [Finkel and Manning, 2009] exist to extract these from text.

Secondly, that while 30% of an article's keywords are inferred and cannot be found within the text without intelligent input, 60% are present in the article's title and first few sentences [Lin and Hovy, 1997], since important facts are generally stated as part of an article's *above the fold* content.

1.5 Evaluation Methods

Shahaf et al. [2012*b*] evaluated their system both for accuracy and with a user study, though as previously discussed they did not evaluate the aesthetic properties of the generated maps. The accuracy evaluation tested whether the system included the most 'important' (as decided by experts) documents in the map.

The user study focused on the strength of the results returned by specific queries, where output was transformed into a structureless list in order for the study to be double-blind against the other methods. The evaluation was performed between-subjects, so background knowledge had to be controlled for. Output was compared with that from Google News and a TDT (Topic Detection and Tracking) method presented in [Nallapati, 2003].

This approach to evaluation is less relevant to my proposed system, since it was actually evaluating the performance of the system in selecting documents based on a query, rather than visualising the documents on a map. In contrast, a visualisation and its usability is precisely the aspect of my process which I would need to evaluate.

The evaluation of TIARA [Liu, Zhou, Pan, Qian, Cai and Lian, 2009] was a more relevant method than the Metro Map evaluation as it was conducted against a baseline system which did not share any of its advanced features, although it was tailored for the same task; email

analysis. A series of questions were asked of participants, who used either TIARA or the baseline system to answer. The response time and accuracy of the participants was recorded, as well as their levels of satisfaction after completing the task. This evaluation used between-subjects designs and therefore required the use of a different dataset for each task, as the nature of the sensemaking means any repetition of the evaluation task on the same data would see participants' performance improve significantly due to recall alone.

1.6 Summary

To recap, this review began with an exploration of the problem of news overload using the findings of The Associated Press and Context-Based Research Group [2008], a field study conducted into the news consumption habits of young people. The findings of the study were then discussed, firstly in the context of four dimensions of information overload [Ho and Tang, 2001, Bergamaschi et al., 2010] and secondly in terms of how they relate to sensemaking; the cognitive process this project aims to support.

Information visualisation was identified as one common approach to both support sensemaking and reduce information overload, so various methods for visualising text-based documents and news articles specifically were presented and compared [Goyal et al., 2013, Havre et al., 2002, Liu, Zhou, Pan, Qian, Cai and Lian, 2009, Singh et al., 2015, Wang et al., 2006] as well as a more in-depth exploration of visual metaphors, in particular timeline, and schematic map visualisations. This section also featured the introduction of the metro map in its original context, before moving on to its use as a metaphor for visualisation in various domains.

Centrally, the work of Shahaf et al. [Shahaf and Guestrin, 2010, Shahaf et al., 2012*b*,*a*, 2013] which is particularly relevant to the aims and objectives of this project was discussed in detail, with an explanation of key metrics (*coherence*, *coverage* and *complexity*) defined in [Shahaf et al., 2012*b*] which are applicable to all graph-based representations of document collections. My main critiques of this body of work were the fixed background corpus and the physical layouts of the drawn maps when evaluated against the criteria defined in [Purchase et al., 1997, Stott et al., 2011].

Next followed a practical overview of methods for transforming news articles into entities with visual or comparative properties, including RSS feed mining, entity recognition, and keyword extraction. Lastly, approaches to experimental design and evaluation by several of the aforementioned studies were discussed in the context of their relevance to the proposed system.

The background, techniques and terminology discussed in this review will be taken forward into the next chapters, as they were all at least partially influential in the design and implementation of the system.

Chapter 2

Requirements

2.1 Project Scope

For the purpose of this dissertation and its timescale, the scope of the project was limited to the development of a system which generates metro maps based on the content of user-specified RSS feeds.

The subject of news fatigue and its possible remedies span multiple domains from data science to cognitive psychology to journalism and content publishing, so while there are many possible techniques in the feasible scope of the project, the practical scope required significant narrowing.

No focus was be given to the content of the RSS feeds themselves, as the standard is sufficiently specified in order for its attributes to be usable without understanding what is represented. Although it is a recognised problem that there is no standardisation for the granularity of RSS feed categories [Liu, Han, Noro and Tokuda, 2009], the only effect varying this would have on the system would be the granularity of the output visualisation.

Likewise, RSS feed discovery and recommendation, while clearly a potential extension to this work from the perspective of improving usability, was not considered.

In respect to the dimensions of information overload [Ho and Tang, 2001] discussed in the previous chapter; while a degree of attention will be given to addressing all four dimensions, the focus will be on information quality–in particular, contextual quality–and fragmentation. Information format, albeit an important facet of overload in general, does not vary significantly between major news publishers. Therefore, while this project may contribute a new unified format for displaying a collection of articles, changing the format or textual content of the articles themselves is not within its scope; any changes which do occur are incidental.

The aim of this work was ultimately the implementation of an end-to-end process for transforming news feeds into metro map schematics; that is to say, none of the techniques themselves are new. The area of interest was the transformation of data from articles to points in physical space on a transit map, and how varying the functions which compose the transformation affect the end result.

2.2 Requirements Gathering

This project was primarily research-based, so while I chose to follow typical software engineering practices by writing a requirements specification for organisational purposes, I did not undertake a formal requirements gathering process with potential users.

Instead, my requirements were derived from the work discussed in my literature review; features of the existing news visualisation system [Shahaf et al., 2012*b*], the transit map aesthetic principles formulated in [Stott, 2008, Stott et al., 2011], and Shneiderman's [1996] InfoVis task taxonomy. Underpinning the technical requirements are the high-level recommendations for news producers specified by The Associated Press and Context-Based Research Group [2008] and Nordenson [2008].

The full requirements specification can be found in Appendices A.1 and A.2, with a discussion on certain conflicts and important decisions later in this chapter.

2.3 Prioritisation

Requirements were assigned priorities using the MoSCoW technique, since the size of the proposed system was not large enough to warrant more granularity in requirement priority. MoSCoW assigns requirements to one of four categories [Waters, 2009];

- **M**ust have - Essential features required for the project to be useful.
- **S**hould have - High value but non-critical features.
- **C**ould have - Desirable features which will be moved out of scope if necessary.
- **W**on't have - Features which have been requested but won't be included.

As the requirements gathering process was based on analysing the findings of other researchers rather than surveying potential users, there were no requirements with a *won't have* modifier.

2.3.1 Categorisation

The operation of the proposed system suggests a natural pipeline of four components through which data will be transformed, with each component comprising some distinct functionality which can be designed, implemented and if necessary modified, in isolation. The components are as follows;

- *Article Retrieval* - The process of parsing an RSS feed and downloading content from the articles it syndicates.
- *Keyword Extraction* - The language processing component, wherein articles are tokenised and their significant keywords are extracted.
- *Graph Building* - The transformation of a collection of articles and their associated keywords into a graph structure, by selecting keywords which best represent the entire corpus. The graph has no physical layout during this stage of the pipeline.

- *Map Drawing* - The generation of a visual representation of the graph structure in the form of a metro map, which the user will interact with.

In addition to the four stages of the pipeline, the system requires an ancillary storage component, to allow processed corpora and their graphs to be imported and exported. In the specification, all functional requirements were grouped according to one of the four stages, to assist in the implementation planning and testing processes.

2.4 Discussion

Central to the importance of the project is the knowledge that news consumers do not subscribe to single RSS feeds; they specifically seek out software which aggregates multiple feeds for convenience [Wang et al., 2006]. Consequentially, F1.1 specifies that the system must accept multiple RSS feeds. However, news preferences are diverse, which gives rise to the potential case where a user specifies multiple feeds which do not share any significant keywords, and articles from one feed are excluded from the visualisation due to their lack of connectivity with the others.

Initially, I considered a requirement for the connectivity [Shahaf et al., 2012*b*] of every metro line to be greater than one; that is, no line should be included in the visualisation unless it intersects with another. However, in the case described above, mutually exclusive RSS feeds could lead to "orphaned" lines which still contain useful content but do not share contextual links to the main body of the map. The stories on these lines should not be ignored by the system, as they most likely would not be ignored by a reader using a traditional RSS reader. My argument is that the lack of connectivity of an article can itself be viewed as metadata on that article, and is not something to be penalised. However, a map exclusively containing orphaned lines is indicative of poor line selection, or–though less likely–a feed containing a set of completely unrelated articles. This highlights a fundamental assumption of the project; that in order to be of use, RSS feeds should contain explicitly related content.

A second issue of deliberation was the specification that stations on the generated maps should not be labelled (F4.7). The use of node labels is logical for a search task, where locating a particular entity or route on a map is the goal. However, when the goal is gaining an insight into the structure of the collection being visualised, labelling each station with the title of the article it represents would only reintroduce the information overload we have been trying to mitigate. The *overview* in the "Overview first" clause of the information seeking mantra [Shneiderman, 1996] is by definition the highest level of abstraction available on each data point, and while in a typical RSS reader this may well be the title, in our system this will not be the case. The context of the articles; i.e. the lines themselves, provide the overview, and the titles instead fall under "details-on-demand."

One requirement which was left deliberately vague was (F4.8); where possible, maps should comply with Stott's [2008] aesthetic criteria for metro map layouts. The lack of strictness at this stage was due to the non-spatial nature of the data being represented; while it is likely that there is an optimal layout for metro maps schematising real transit networks due to the geographic positioning of stations on those networks, there is no "sensible" underlying topology to our data. It was unknown at this stage whether we would be able to enforce strict layout criteria without significant pruning of the data.

Chapter 3

Design and Implementation

3.1 Introduction

During the requirements gathering process, four distinct components of the system were identified which form a pipeline of execution; *Article Retrieval*, *Keyword Extraction*, *Graph Building* and *Map Drawing*. Crucially, each is specified as being modular, allowing both for flexible extensibility and for alternative implementations to be tested directly against each other without requiring changes to the other components.

This chapter further decomposes these components to provide a detailed overview of the methodology used to implement them, followed by a discussion of the significant challenges and successes which arose during development.

3.2 Code Reuse

I developed the design of the system in Python 2.7 and JavaScript 1.7 using various open-source libraries and APIs for its ancillary functionality. The most notable are detailed below and discussed in context in the following sections.

- **FeedParser**[1]: A Python module for downloading and parsing RSS feeds.
- **Newspaper**[2]: A Python library for downloading and extracting content and metadata from online articles.
- **lxml**[3]: A Python library for generating, parsing and manipulating XML and HTML.
- **NLTK (The Natural Language Toolkit)**[4]: A Python library for natural language processing and analysis.
- **Google Knowledge Graph Search**[5]: The API for Google's knowledge base, which returns structured semantic search results.
- **D3.js**[6]: A JavaScript library for creating and manipulating interactive web visualisations and their underlying data.

[1] `http://pythonhosted.org/feedparser`
[2] `http://newspaper.readthedocs.io`
[3] `http://lxml.de`
[4] `http://www.nltk.org`
[5] `https://developers.google.com/knowledge-graph`
[6] `http://d3js.org`

3.3 High-Level Design

Figure 3.1 outlines the decomposition of the four pipeline into their most important subtasks.

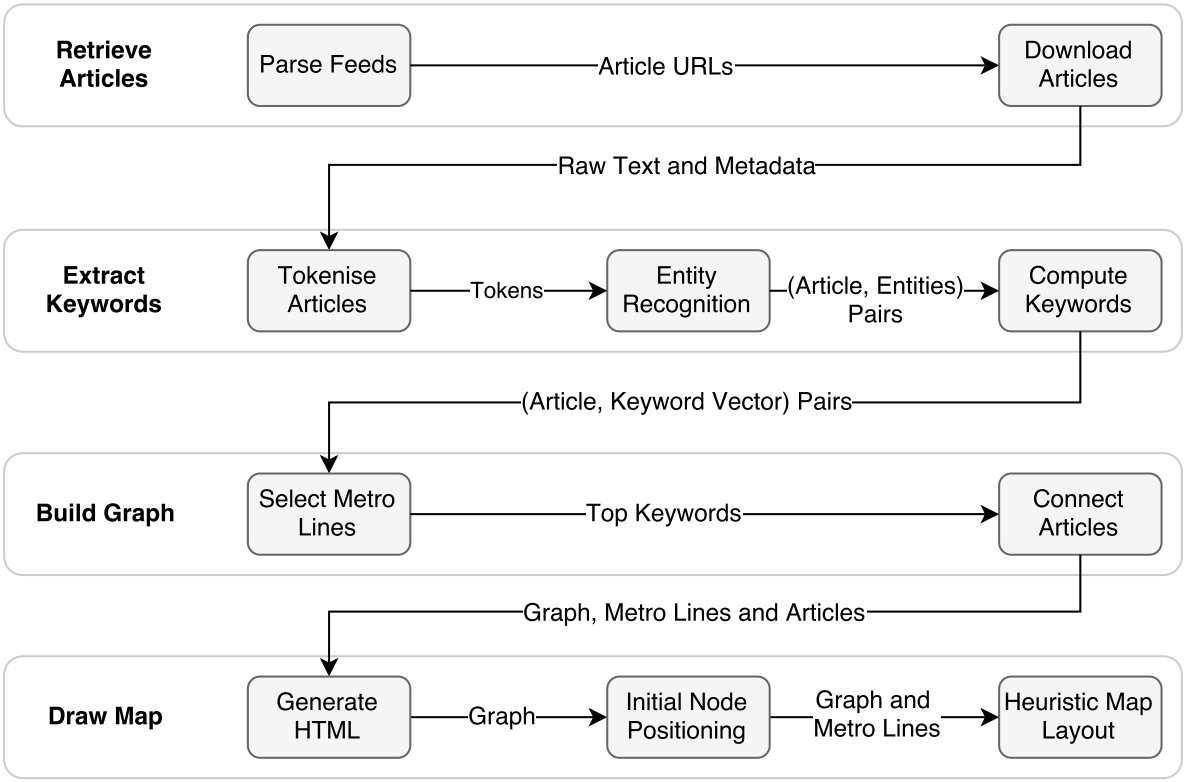


Figure 3.1: A conceptual model of data flow between components of the system.

Not included in Figure 3.1 is the boundary between the *Build Graph* and *Draw Map* stages of the pipeline, which marks the transition from the run-time Python which extracts and processes the article data, to JavaScript which only runs once the visualisation is opened in the browser; the entirety of the *Draw Map* stage. The boundary takes the form of an intermediate stage between *Build Graph* and *Draw Map*, where the topological map and the article data required for the visualisation are serialised to JSON[7].

3.4 Article Retrieval

The first stage of article retrieval is the parsing of RSS feeds, in order for article URLs and metadata to be extracted. The Python library FeedParser was used, as at the time of writing it provided the best support for RSS 0.9x, 1.0 and 2.0. For each item in the feed, the parsing process extracts a link to each article, the name of the channel, and the parsed publish date, but it will also attempt to extract the author name if the `author` attribute is found.

Once the feed data has been extracted, it is used to construct an instance of `ArticleCollection`, which acts a wrapper around the contents of one or more feeds and provides the mechanism

[7] JavaScript Object Notation; `http://www.json.org`

necessary to perform corpus-wide queries such as calculating inverse document frequency for TF-IDF, and proportional document frequency for TF-PDF. Individual articles are downloaded using the Newspaper library, however due to the requirement that `ArticleCollection` must be serialisable with cPickle to provide intermediate cache format, it was necessary to create a new `Article` class which unpacks the necessary members of `Newspaper.Article` and discards the XML which prevents instances of `Newspaper.Article` from being serialisable. The `Article` encapsulates the functions for computing article-specific terms such as term-frequency for TF-IDF, and the term-weighting component of TF-PDF.

3.4.1 Sanitisation

An early problem encountered was the embedding of unrelated content into the body of articles by news publishers in order to encourage readers to seek out additional content from the same website. This content typically takes the form of a gallery or list of images and article titles with brief summaries or subheads (see Listing 3.1).

```
1 <div id='article'>
2   <h1> Article Title </h1>
3   <p> ... </p>
4   <p> ... </p>
5   <div id='gallery'>
6     <h1> Related Articles </h1>
7     <p> ... </p>
8   </div>
9 </div>
```

Listing 3.1: Unrelated content nested into the body of an article

If this content was left and processed as part of an article, it would result in keywords being falsely detected in the articles when in fact they were only present in the form of a link to unrelated content. It was therefore necessary to sanitise the article HTML before tokenisation and keyword extraction. The semantic structure of HTML made this a simpler task; any child elements of the element containing the body text of the article are removed (using lxml) unless they are a paragraph or heading.

The result of this pre-processing stage is an `ArticleCollection` containing one or more serialisable `Article`s from one or more RSS feeds, where each `Article` contains extracted and sanitised but unparsed data, and associated metadata.

3.5 Keyword Extraction

The keyword extraction stage begins with the process of tokenising the body text of every article. Tokenisation here has three substages, all of which were implemented using NLTK (The Natural Language Toolkit for Python), and which are as follows:

1. Sentence segmentation: Split the text into a list of sentences and remove sentence punctuation.

2. Tokenisation: Split each sentence into a list of individual words and remove both whitespace and clause punctuation.

3. Part-of-Speech (POS) tagging: Categorise each token according to its lexical class, e.g. adjective. This more of an extension to the tokenisation process than a part of it, but it is performed directly after tokenisation and is necessary for the next stage of processing.

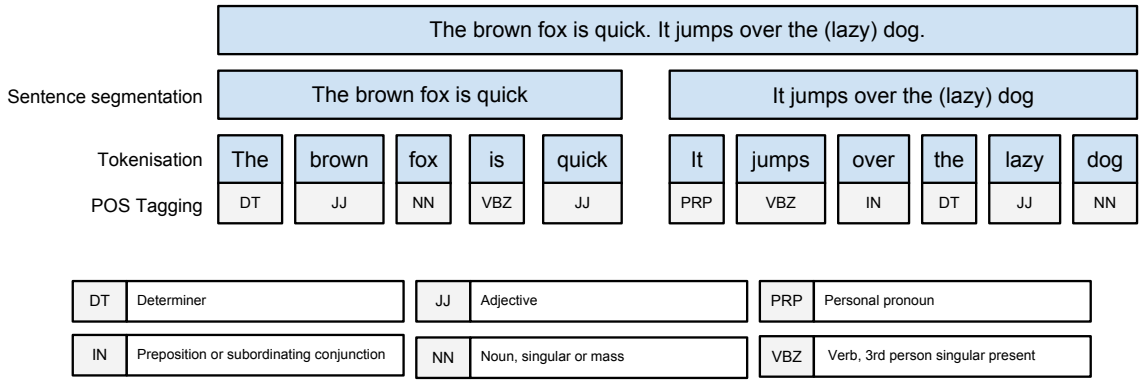


Figure 3.2: Stepping through the tokenisation process

3.5.1 Named Entity Recognition

The second half of the keyword extraction process is exclusively concerned with named entities within articles. This is because, intuitively, the names of people, places, events companies and other *things* form a set of strong candidate keywords. Restricting the candidates to entities alone reduces the search space by an order of magnitude when using frequency-based methods for keyword extraction, and bypasses the need for other natural language processing tasks such as stop-word removal and lemmatisation.

Once tokens have been POS tagged, named-entity chunking can be performed, again using NLTK. This process groups tokens into contiguous and non-overlapping *chunks*, where each chunk is a named entity; typically a proper noun or some other noun phrase.

It is possible for chunks to contain other chunks (consider the chunk 'Bank of England', which contains the chunk 'England'), but this is typically undesirable for entity recognition – where we desire specificity – so after chunking the tokens, we flatten the chunk structure so chunks cannot be any further decomposed.

At this stage, we have a list of chunks, each of which will be an eventual candidate for becoming a metro line. However, the chunks require some further processing before we can determine whether or not they are significant keywords.

Name Disambiguation: Substring Matching

It is common stylistic practice in journalism to refer to the subjects of articles by their surnames. However, to avoid any confusion, the full names of those mentioned will often

appear in the title or first few paragraphs of the article. If keyword strength is determined by frequency, then regardless of whether we use TF-IDF or TF-PDF, we need every occurrence of an entity to refer to that entity by the same name; preferably the most specific, which is typically the longest.

During name disambiguation, we only consider entities with more than one mention in the source text as candidates. This is because the chunking process can produce false positives by combining unrelated adjectives with valid chunks, but the likelihood of the same false positive being produced twice or more in the same article is low.

Let P denote the set of entities with more than one occurrence in the source article; S:

$$P = \{e \mid \text{occurences}(e, S) > 1\}.$$

$\forall (e_1, e_2) \in \{P \times P \mid \text{length}(e_1) < \text{length}(e_2)\}$, if e_1 is a substring of e_2, we call e_1 an *alias* of e_2. Let A be the set of alias pairs in S;

$$A = \{(a, e) \mid a \text{ is an alias of } e\}$$

We require the first term of every pair in A to be unique, but there is no such constraint for the second term (e); this allows multiple aliases to map to the same entity. For example,

$$\{(\text{'Zuckerberg'}, \text{'Mark Zuckerberg'}), (\text{'Zuck'}, \text{'Mark Zuckerberg'})\}$$

would be unambiguous and therefore valid, but

$$\{(\text{'Mark'}, \text{'Mark Zuckerberg'}), (\text{'Mark'}, \text{'Zuckerberg'})\}$$

would not be. Let U be the set of unambiguous alias-entity pairs in S:

$$U = \{(a, e) \mid (a, e) \in A \cap \forall (e_1, e_2) \in A,\ a = e_1 \implies e = e_2\}$$

We have a reasonable degree of confidence that for every alias-entity pair $(a, e) \in U$, occurrences of a in the source text can be replaced by e, making e a stronger candidate for keyword detection. Pairs in $(a, e) \in A \setminus U$, that is, aliases which could map to multiple entities in the source, are disregarded at this stage and left unchanged. A more sophisticated method of name disambiguation could address this.

Algorithm 1 illustrates how, given a list of entities, we can find the unambiguous pairs deterministically in $\mathcal{O}(n^2)$ time.

Entity Disambiguation

Entity disambiguation describes the process of determining the identity of entities in a body of text. Performing this process on the sentence 'The UK has voted to leave the EU,' should identify 'UK' as The United Kingdom and 'EU' as The European Union. As advanced methods for entity disambiguation are both complex and computationally expensive, I did not attempt to implement a formal method for this.

Algorithm 1: Finding unambiguous alias-entity pairs

Data: *names*: a list of recognised entities
Result: U: the set of unambiguous pairs in *names*

```
1  U ⇐ {};
2  foreach e1, e2 ∈ (names × names) do
3  |  if len(e1) > len(e2) then
4  |  |  swap(e1, e2);
5  |  end
6  |  if e1 ∈ e2 then                          /* If e1 is a substring of e2 */
7  |  |  if e1 ∉ U then
8  |  |  |  U[e1] ⇐ e2 ;                      /* (e1, e2) are a candidate pair */
9  |  |  else
10 |  |  |  delete U[e1] ;                    /* e1 is now ambiguous, so remove it */
11 |  |  end
12 |  end
13 end
```

Instead, the list of recognised entities are queried against Google's publicly accessible Knowledge Graph API, which returns a list of potential results as Schema.org[8] types. Knowledge Graph is the service which replaced Freebase in 2015, and currently contains over 70 billion facts [@jeffjarvis Twitter account, 2016].

Each result has a score attributed to it by Knowledge Graph, which is an indicator the strength of the match between the entity and the original query. Results are sorted by descending score; the higher the `resultScore`, the better the match. Although there is no defined upper limit for this value, comparing the scores of the top two results for a query can provide a measure of certainty, for all but particularly esoteric or unknown entities.

We specify a threshold $\frac{1}{t}$, which is roughly proportional to the likelihood of accepting a false positive match. Then, if dividing the score of the first result by the score of the second yields a number greater than $\frac{1}{t}$, we accept the match. Provisionally we set $t = 0.5$; a discussion of how I arrived at this value is provided in the next section.

Using a knowledge base for disambiguation also inadvertently solves another problem I encountered while parsing articles, this time as a result the expositional style of journalistic writing. While keywords are typically nouns or noun phrases, they can also appear in the form of denominal adjectives. These adjectives are derived from nouns; e.g. 'French' implies 'France' might be a keyword. Denominal adjectives are not amenable to traditional stemming or lemmatising, but querying Knowledge Graph for 'French' returns a top match of 'France' with a `resultScore` of 432.42807; more than four times larger than the next result.

Given a list of entities E, and top two results of a Knowledge Graph query for each $e \in E$; $R_e(1)$ and $R_e(2)$ with scores $S_e(1)$ and $S_e(2)$ respectively, our aim is to return a set of pairs mapping zero or more entities in E to their disambiguated forms;

[8] http://schema.org is an online hierarchy of types managed by W3C (The World Wide Web Consortium)

$$K = \left\{ (e, R_e(1)) \mid \frac{S_e(1)}{S_e(2)} > \frac{1}{t} \right\}$$

Algorithm 2 describes this process. For higher values of t, the likelihood of accepting a false positive increases, and for lower values, the likelihood of accepting a false negative increases. For corpora which do not contain references to public figures and place names, choice of t should be empirically tuned against the prevalence of the expected entities.

Algorithm 2: Entity disambiguation with Knowledge Graph

Data: *names*: a list of recognised entities
t: the acceptance threshold for results, default = 0.5
Result: K: a set of disambiguation mappings for elements in *names*

```
1  K ⇐ {};
2  T ⇐ 1 ÷ t;
3  foreach e ∈ names do
4  |   results ⇐ KnowledgeGraphResults(e);
5  |   if len(results) > 1 then
6  |   |   if results[0].score ÷ results[1].score > T then
7  |   |   |   K[e] = results[0].name;
8  |   |   end
9  |   end
10 end
```

It is unclear what should become of queries that Knowledge Graph only returns a single result for, but in this implementation they are ignored.

Determining Optimal Values for t

Empirically, I found the best values for t are in the range $0.35 < t < 0.65$, meaning we accept top results which are at least 1.5x-3x higher than the next best candidate. To determine this, I logged all the disambiguation pairs found in 20 articles from the BBC Politics RSS feed, letting t range over $\{0.3, 0.5, 0.7, 0.9\}$. I then manually classified the pairs as either true or false positives.

Figure 3.3 shows the results of this investigation, with total pairs plotted alongside the percentage of those pairs which were false positives.[9] Although we see a reduction in false positives for smaller values of t, the trend-line illustrates that this is a game of diminishing returns; reducing t from 0.3 to 0.1 only reduces the false positives by 4.34%, but it leads to 43 fewer pairs being identified.

There is a clear trade-off between the accuracy of the disambiguation and the number of false negatives we discard, but the cost of false positives in this case is less than the cost of false negatives. While a false positive could result in unrelated entities appearing as keywords

[9]Since it is both extensive and tangential to the focus of the project, raw data for this table can be found at `http://bit.ly/DisambiguationThresholding`.

for certain articles, the likelihood of the same false positive appearing with high frequency in enough articles to result in an erroneous metro line is incredibly low. In contrast, the cost of disregarding a false negative could result in articles being left off certain metro lines altogether. It is for this reason that we don't simply choose the value of t which yields the minimum ratio of false positives.

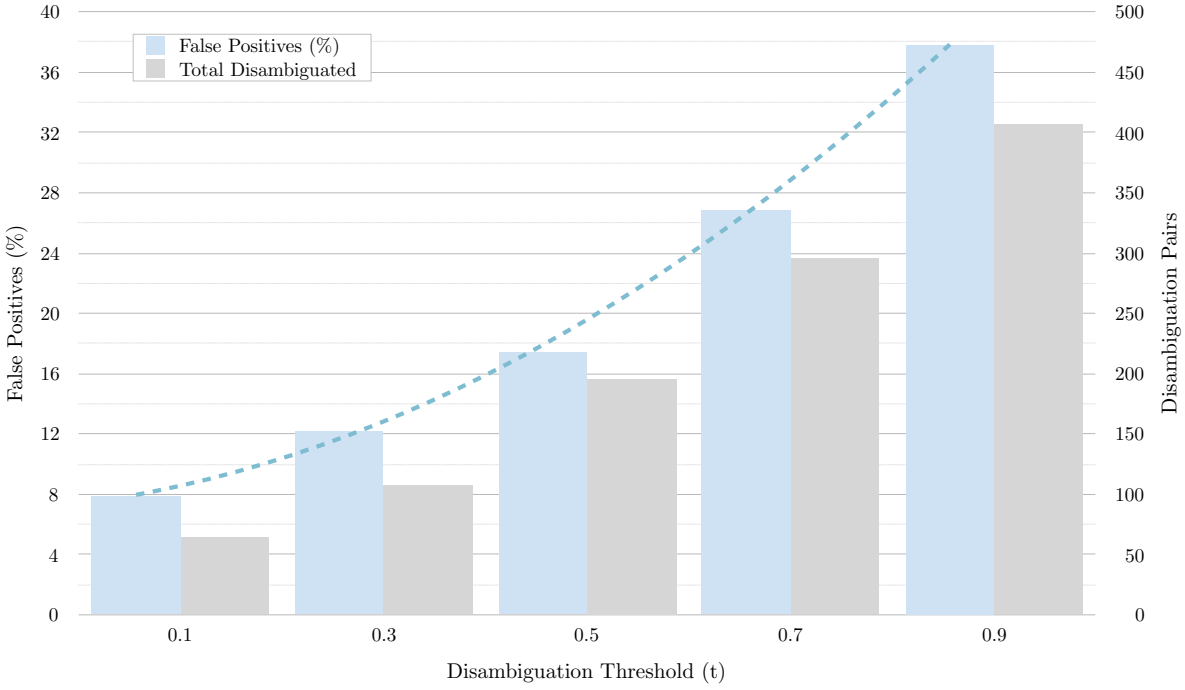


Figure 3.3: Disambiguation threshold against pairs found and false positives (%)

3.5.2 Keyword Ranking

Given a set of disambiguated entities for every article, the union of which forms a set of candidates keywords or *metro lines* for the collection, we must now determine which keywords are the most relevant. To do this, the two methods described in Chapter 1 will be revisited; TF-IDF [Sparck Jones, 1972] and TF-PDF [Bun and Ishizuka, 2002]. Both were implemented in the system so they could be compared directly against each another.

From an implementation perspective, the main difference between these two algorithms are the level at which they operate. TF-IDF is calculated on a per-article basis, returning a vector of an article's keywords and their corresponding scores.

In contrast, TF-PDF is specified at corpus level, where there are one or more channels within the corpus. In our case, channels can be conveniently defined as RSS feeds, to correct the bias which could arise from all articles within a channel containing a specific keyword which is then identified as significant due to the relatively small size of that channel against the rest of the corpus. Similarly to TF-IDF however, it produces a vector of pairs of keywords and their scores.

For both equations, the advantage to restricting the set of candidates to named entities

is apparent. Figure 3.4 shows the number of tokens against the percentage of extracted entities for 40 articles from the BBC's Politics RSS Feed.[10] The interquartile range is shown, illustrating that 50% of articles had entities comprising 5.4%-8.2% of their tokens after stop-word removal. With the mean equal to 6.65%, the implications of this are a search space which can be reduced by a factor of more than ten. Although performance optimisation was not specified in the aims of the project, gains of this nature are still significant.

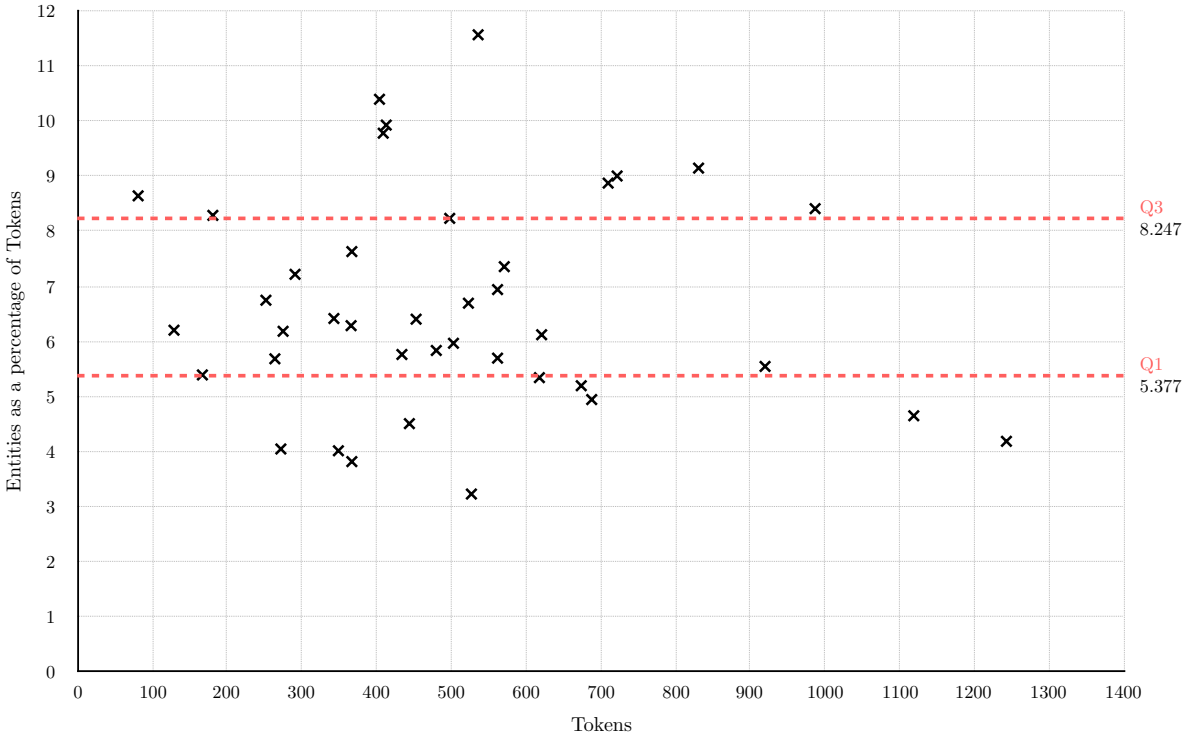


Figure 3.4: Entities as a percentage of Tokens across 40 BBC Politics articles.

3.6 Graph Building

3.6.1 Choosing Lines

3.6.2 Connecting Articles

3.7 Map Drawing

3.7.1 Initial Node Positioning

- D3.js for force directed layout
- Initial force parameters; adjusting based on Stott [2008], Stott et al. [2011]

[10] http://feeds.bbci.co.uk/news/politics/rss.xml (Accessed: 27/02/2017, Full data in Appendix B.1)

3.7.2 Heuristic Layout

- Stott [2008], Stott et al. [2011] for snap()
- Repeated averaging along lines for sensible octilinearity
- Straightening the ends of lines

Chapter 4

Results and Discussion

Chapter 5

Empirical Evaluation

Chapter 6

Conclusions

Bibliography

Associated Press and Context-Based Research Group [2008], 'A new model for news: Studying the deep structure of young adult news consumption'. Accessed: 26/10/2016.
URL: *http://manuscritdepot.com/edition/documents-pdf/newmodel.pdf*

Barzilay, R., McKeown, K. R. and Elhadad, M. [1999], Information fusion in the context of multi-document summarization, *in* 'Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics', Association for Computational Linguistics, pp. 550–557.

Bergamaschi, S., Guerra, F. and Leiba, B. [2010], 'Guest editors' introduction: information overload', *IEEE Internet Computing* **14**(6), 10–13.

Binh Tran, G. [2013], Structured summarization for news events, *in* 'Proceedings of the 22nd International Conference on World Wide Web', WWW '13 Companion, ACM, New York, NY, USA, pp. 343–348.
URL: *http://doi.acm.org/10.1145/2487788.2487940*

British Broadcasting Corporation [2013], 'Tube 150th anniversary: How the underground map evolved'. Accessed: 4/1/2017.
URL: *http://www.bbc.co.uk/news/uk-england-london-20943525*

Bruns, A., Highfield, T. and Lind, R. A. [2012], 'Blogs, twitter, and breaking news: The produsage of citizen journalism', *Produsing theory in a digital world: The intersection of audiences and production in contemporary theory* **80**(2012), 15–32.

Bun, K. K. and Ishizuka, M. [2002], 'Topic extraction from news archive using tf-pdf algorithm', *Proceedings of the 3rd International Conference on Web Information Systems Engineering* .

Burkhard, R. A. [2004], Learning from architects: the difference between knowledge visualization and information visualization, *in* 'Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on', IEEE, pp. 519–524.

Carpena, P., Bernaola-Galván, P., Hackenberg, M., Coronado, A. and Oliver, J. [2009], 'Level statistics of words: Finding keywords in literary texts and symbolic sequences', *Physical Review E* **79**(3), 035102.

Dredze, M., McNamee, P., Rao, D., Gerber, A. and Finin, T. [2010], Entity disambiguation for knowledge base population, *in* 'Proceedings of the 23rd International Conference on Computational Linguistics', COLING '10, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 277–285.
URL: *http://dl.acm.org/citation.cfm?id=1873781.1873813*

Eppler, M. J. [2004], Visuelle kommunikationder einsatz von graphischen metaphern zur optimierung des wissenstransfers, *in* 'Wissenskommunikation in Organisationen', Springer, pp. 13–31.

Eppler, M. J. [2006], 'A comparison between concept maps, mind maps, conceptual diagrams, and visual metaphors as complementary tools for knowledge construction and sharing', *Information visualization* **5**(3), 202–210.

Ferrari, D. and Mezzalira, L. [1970], 'A computer-aided approach to integrated circuit layout design', *Computer-Aided Design* **2**(2), 19–23.

Finkel, J. R. and Manning, C. D. [2009], Nested named entity recognition, *in* 'Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1', Association for Computational Linguistics, pp. 141–150.

Fischer, G. and Stevens, C. [1991], Information access in complex, poorly structured information spaces, *in* 'Proceedings of the SIGCHI Conference on Human Factors in Computing Systems', CHI '91, ACM, New York, NY, USA, pp. 63–70.

Garg, A. and Tamassia, R. [1994], On the computational complexity of upward and rectilinear planarity testing, *in* 'International Symposium on Graph Drawing', Springer, pp. 286–297.

@GoogleTrends Twitter account [2016], 'Tweet: *+250% spike in "what happens if we leave the EU" in the past hour*', `http://twitter.com/GoogleTrends/status/746137920940056578`. Accessed: 27/10/2016.

Goyal, R., Malla, R., Bagchi, A., Mehta, S. and Ramanath, M. [2013], Esthete: A news browsing system to visualize the context and evolution of news stories, *in* 'Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management', CIKM '13, ACM, New York, NY, USA, pp. 2529–2532.
URL: *http://doi.acm.org/10.1145/2505515.2508208*

Hargreaves, I., Thomas, J., Commission, I. T. and Commission, G. B. B. S. [2002], *New news, old news: an ITC and BSC research publication*, Independent Television Commission.
URL: *https://books.google.co.uk/books?id=VZ-hPAAACAAJ*

Havre, S., Hetzler, E., Whitney, P. and Nowell, L. [2002], 'Themeriver: Visualizing thematic changes in large document collections', *IEEE transactions on visualization and computer graphics* **8**(1), 9–20.

Ho, J. and Tang, R. [2001], Towards an optimal resolution to information overload: An infomediary approach, *in* 'Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work', GROUP '01, ACM, New York, NY, USA, pp. 91–96.
URL: *http://doi.acm.org/10.1145/500286.500302*

Hochmair, H. [2009], 'The influence of map design on route choice from public transportation maps in urban areas', *The Cartographic Journal* **46**(3), 242–256.

Holton, A. E. and Chyi, H. I. [2012], 'News and the overloaded consumer: Factors influencing information overload among news consumers', *Cyberpsychology, Behavior, and Social Networking* **15**(11), 619–624.

Husin, H. S., Thom, J. A. and Zhang, X. [2014], Analysing user access to an online newspaper, *in* 'Proceedings of the 2014 Australasian Document Computing Symposium', ADCS '14, ACM, New York, NY, USA, pp. 77:77–77:80.

@jeffjarvis Twitter account [2016], 'Tweet: *Google knowledge graph has more than 70 billion facts about people, places, things. + language, image, voice translation*', `https://twitter.com/jeffjarvis/status/783338071316135936`. Accessed: 20/02/2017.

Lin, C.-Y. and Hovy, E. [1997], Identifying topics by position, *in* 'Proceedings of the fifth conference on Applied natural language processing', Association for Computational Linguistics, pp. 283–290.

Lipton, R. J., North, S. C. and Sandberg, J. S. [1985], A method for drawing graphs, *in* 'Proceedings of the first annual symposium on Computational geometry', ACM, pp. 153–160.

Liu, B., Han, H., Noro, T. and Tokuda, T. [2009], Personal news rss feeds generation using existing news feeds, *in* 'International Conference on Web Engineering', Springer, pp. 419–433.

Liu, S., Zhou, M. X., Pan, S., Qian, W., Cai, W. and Lian, X. [2009], Interactive, topic-based visual text summarization and analysis, *in* 'Proceedings of the 18th ACM Conference on Information and Knowledge Management', CIKM '09, ACM, New York, NY, USA, pp. 543–552.

Nallapati, R. [2003], Semantic language models for topic detection and tracking, *in* 'Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Proceedings of the HLT-NAACL 2003 student research workshop-Volume 3', Association for Computational Linguistics, pp. 1–6.

Nguyen, P. H., Xu, K., Walker, R. and Wong, B. [2014*a*], 'Timesets: timeline visualization for sensemaking'.

Nguyen, P. H., Xu, K., Walker, R. and Wong, B. W. [2014*b*], Schemaline: timeline visualization for sensemaking, *in* 'Information Visualisation (IV), 2014 18th International Conference on', IEEE, pp. 225–233.

Nordenson, B. [2008], 'Overload! Journalism's battle for relevance in an age of too much information', *Columbia Journalism Review* **47**(4), 30.

O'donnell, A. M., Dansereau, D. F. and Hall, R. H. [2002], 'Knowledge maps as scaffolds for cognitive processing', *Educational psychology review* **14**(1), 71–86.

Old, L. J. [2002], Information cartography: Using gis for visualizing nonspatial data, *in* 'Proceedings of the ESRI International Users Conference', Citeseer.

O'Shea, M. and Levene, M. [2011], 'Mining and visualising information from rss feeds: a case study', *International Journal of Web Information Systems* **7**(2), 105–129.

Pentina, I. and Tarafdar, M. [2014], 'From "information" to "knowing": Exploring the role of social media in contemporary news consumption', *Computers in Human Behavior* **35**, 211–223.

Pera, M. S. and Ng, Y.-K. [2008], 'Utilizing phrase-similarity measures for detecting and clustering informative rss news articles', *Integrated Computer-Aided Engineering* **15**(4), 331–350.

Phuvipadawat, S. and Murata, T. [2010], Breaking news detection and tracking in twitter, *in* 'Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on', Vol. 3, IEEE, pp. 120–123.

Purcell, K., Rainie, L., Mitchell, A., Rosenstiel, T. and Olmstead, K. [2010], 'Understanding the participatory news consumer', *Pew Internet and American Life Project* **1**, 19–21.

Purchase, H. [1997], Which aesthetic has the greatest effect on human understanding?, *in* 'International Symposium on Graph Drawing', Springer, pp. 248–261.

Purchase, H. C., Cohen, R. F. and James, M. I. [1997], 'An experimental study of the basis for graph drawing algorithms', *J. Exp. Algorithmics* **2**.
URL: *http://doi.acm.org/10.1145/264216.264222*

Rennison, E. [1994], Galaxy of news: An approach to visualizing and understanding expansive news landscapes, *in* 'Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology', UIST '94, ACM, New York, NY, USA, pp. 3–12.
URL: *http://doi.acm.org/10.1145/192426.192429*

Rosen, J. [2008], 'National explainer: A job for journalists on the demand side of news', `http://archive.pressthink.org/2008/08/13/national_explain.html`. Accessed: 2/11/2016.

Russell, D. M., Slaney, M., Qu, Y. and Houston, M. [2006], Being literate with large document collections: Observational studies and cost structure tradeoffs, *in* 'Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)', Vol. 3, IEEE, pp. 55–55.

Salton, G. and Buckley, C. [1988], 'Term-weighting approaches in automatic text retrieval', *Information processing & management* **24**(5), 513–523.

Schick, A. G., Gordon, L. A. and Haka, S. [1990], 'Information overload: A temporal approach', *Accounting, Organizations and Society* **15**(3), 199–220.

Shahaf, D. and Guestrin, C. [2010], Connecting the dots between news articles, *in* 'Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 623–632.

Shahaf, D., Guestrin, C. and Horvitz, E. [2012*a*], Metro maps of science, *in* 'Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '12, ACM, New York, NY, USA, pp. 1122–1130.
URL: *http://doi.acm.org/10.1145/2339530.2339706*

Shahaf, D., Guestrin, C. and Horvitz, E. [2012*b*], Trains of thought: Generating information maps, *in* 'Proceedings of the 21st International Conference on World Wide Web', WWW '12, ACM, New York, NY, USA, pp. 899–908.
URL: *http://doi.acm.org/10.1145/2187836.2187957*

Shahaf, D., Yang, J., Suen, C., Jacobs, J., Wang, H. and Leskovec, J. [2013], Information cartography: creating zoomable, large-scale maps of information, *in* 'Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 1097–1105.

Shneiderman, B. [1996], The eyes have it: A task by data type taxonomy for information visualizations, *in* 'Visual Languages, 1996. Proceedings., IEEE Symposium on', IEEE, pp. 336–343.

Singh, J., Anand, A., Setty, V. and Anand, A. [2015], Exploring long running news stories using wikipedia, *in* 'Proceedings of the ACM Web Science Conference', WebSci '15, ACM, New York, NY, USA, pp. 57:1–57:2.
URL: *http://doi.acm.org/10.1145/2786451.2786489*

Sparck Jones, K. [1972], 'A statistical interpretation of term specificity and its application in retrieval', *Journal of documentation* **28**(1), 11–21.

Stott, J. [2008], Automatic layout of metro maps using multicriteria optimisation, PhD thesis, University of Kent. Accessed: 11/11/2016.
URL: *http://www.jstott.me.uk/thesis/thesis-final.pdf*

Stott, J., Rodgers, P., Martinez-Ovando, J. C. and Walker, S. G. [2011], 'Automatic metro map layout using multicriteria optimization', *IEEE Transactions on Visualization and Computer Graphics* **17**(1), 101–114.

Strong, D. M., Lee, Y. W. and Wang, R. Y. [1997], 'Data quality in context', *Communications of the ACM* **40**(5), 103–110.

Tamassia, R. [1987], 'On embedding a graph in the grid with the minimum number of bends', *SIAM Journal on Computing* **16**(3), 421–444.

Teitler, B. E., Lieberman, M. D., Panozzo, D., Sankaranarayanan, J., Samet, H. and Sperling, J. [2008], Newsstand: a new view on news, *in* 'Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems', ACM, p. 18.

Transport For London [2014], 'Harry Beck's Tube map', `https://tfl.gov.uk/corporate/about-tfl/culture-and-heritage/art-and-design/harry-becks-tube-map`. Accessed: 18/1/2017.

Wang, T., Yu, N., Li, Z. and Li, M. [2006], nReader: Reading News Quickly, Deeply and Vividly, *in* 'CHI '06 Extended Abstracts on Human Factors in Computing Systems', CHI EA '06, ACM, New York, NY, USA, pp. 1385–1390.

Waters, K. [2009], 'Prioritization using moscow', *Agile Planning* **12**.

Weick, K. E., Sutcliffe, K. M. and Obstfeld, D. [2005], 'Organizing and the process of sensemaking', *Organization science* **16**(4), 409–421.

Wise, J. A., Thomas, J. J., Pennock, K., Lantrip, D., Pottier, M., Schur, A. and Crow, V. [1995], Visualizing the non-visual: Spatial analysis and interaction with information from text documents, *in* 'Proceedings of the 1995 IEEE Symposium on Information Visualization', INFOVIS '95, IEEE Computer Society, Washington, DC, USA, pp. 51–.

Yen, J.-C., Lee, C.-Y., Chen, I. et al. [2012], 'The effects of image-based concept mapping on the learning outcomes and cognitive processes of mobile learners', *British Journal of Educational Technology* **43**(2), 307–320.

Yi, J. S., Kang, Y.-a., Stasko, J. T. and Jacko, J. A. [2008], Understanding and characterizing insights: How do people gain insights using information visualization?, *in* 'Proceedings of the 2008 Workshop on BEyond Time and Errors: Novel evaLuation Methods for Information Visualization', BELIV '08, ACM, New York, NY, USA, pp. 4:1–4:6.
URL: *http://doi.acm.org/10.1145/1377966.1377971*

Ziemkiewicz, C. and Kosara, R. [2009], Preconceptions and individual differences in understanding visual metaphors, *in* 'Computer Graphics Forum', Vol. 28, Wiley Online Library, pp. 911–918.

Appendix A

Full Requirements Specification

A.1 Functional Requirements

1 Article Retrieval

Description

The article retrieval component of the system will accept the URL of one or more RSS feeds, collect links to the articles referenced and parse the text of those articles for further processing.

Functional Requirements

F1.1 The system must accept any standard XML document compliant with the RSS 2.0 specification[1], i.e. it should not be specific to any particular news provider

F1.2 The system must be able to extract a specified number of articles from an RSS feed, in reverse chronological order.

F1.3 The system must be able to download the textual content and/or raw HTML for each article.

F1.4 The system should accept multiple RSS feeds from one or more news provider(s) and merge their content into one collection.

F1.5 The system could verify new article URLs against the URLs of imported articles to ensure no articles are duplicated.

2 Keyword Extraction

Description

The keyword extraction stage will tokenise the parsed article text and determine a set of significant keywords for each article individually.

[1] http://cyber.harvard.edu/rss/rss.html

Functional Requirements

F2.1 The system must tokenise articles in order to perform basic natural language processing such as stop-word extraction and lemmatising.

F2.2 The system must implement a method for keyword extraction, and calculate a corresponding measure of relative importance for each keyword such as TF-IDF.

F2.3 The system could attempt to combine keywords it considers equivalent (e.g. *UK* and *United Kingdom*) to form stronger keyword matches between or within articles.

F2.4 The system could use external services (e.g. Google's Knowledge Graph API[2]) to query any extracted keywords, in order to gain further insight or perform entity disambiguation [Dredze et al., 2010].

3 Graph Building

Description

This process involves determining a set of corpus keywords from the union of all the articles' keywords to form connected paths of edges (*lines*), and fitting a maximal number of articles into the resulting graph.

Functional Requirements

F3.1 The system must analyse the keywords extracted from all articles in a corpus to choose a set of the n most significant topics, where n is either predetermined or user-specified.

F3.2 The system must use the extracted topics and the publish dates (which form a natural ordering of nodes) of the articles to form a directed graph, with articles as vertices and common topic storylines as edges.

F3.3 The system should accept a user-specified topic or list of topics to include or exclude from the graph.

F3.4 The system should choose topics which are specific to some but not all articles in the collection, so as to avoid highly correlated topic keywords.

F3.5 The system could support exporting generated graphs in a graph description language e.g. DOT[3] or GraphML[4].

F3.6 The system could attempt to combine keywords to form topics if it considers them highly correlated.

F3.7 The system could attempt to maximise the coverage of the topic selection, i.e. maximise the number of articles covered by a given set of keywords.

[2] http://www.google.com/intl/es419/insidesearch/features/search/knowledge.html
[3] http://www.graphviz.org/content/dot-language
[4] http://graphml.graphdrawing.org

4 Map Drawing

Description

The visualisation component will generate an interactive visualisation of the graph which can be used to explore the corpus as a whole and drill-down to the individual article level.

Functional Requirements

F4.1 The system must provide the capability for users to visualise the graph structures it generates using any HTML5 compliant web browser.

F4.2 Metro lines must each be drawn in a different colour which contrasts that of other lines.

F4.3 The visualisation must include a key mapping metro line names to their colours on the map.

F4.4 Stations must be drawn with common (non-unique) symbols, with a distinction made for interchange stations.

F4.5 The system must provide drill-down details for nodes, e.g. by providing a hyperlink to the original article or embedding static content from each article within the visualisation itself to provide a preview.

F4.6 The maps generated should be readable by ensuring nodes and edges do not overlap with each other where possible.

F4.7 In support of F4.6, to improve the readability of the map, nodes should not be labelled with article titles.

F4.8 The maps generated should adhere (where possible) to the metro map aesthetics defined in [Stott, 2008, Stott et al., 2011] to preserve the familiarity of the metaphor.

F4.9 The system could allow some degree of interactive customisation which does not change the underlying structure of the graph, such as dragging nodes or changing attributes including colour.

5 Storage and Persistence

Description

This component of the system is responsible for saving and importing previously downloaded corpora and reconstructing their graphs.

Functional Requirements

F5.1 The system must support the importing/exporting of graph and article data in an intermediate data form, in order to fully reconstruct graphs it had previously created.

F5.2 By default, the articles collected by each run of the system must be treated as a new corpus so keyword ranking is deterministic for any given feed.

A.2 Nonfunctional Requirements

1 Security

The system will not require any kind of authentication to use, and will only stores data which is publicly available. As there are no security regulations which govern its usage, security is not a critical consideration and there are only two associated requirements.

NF1.1 The system will not collect any data during installation and usage without obtaining consent from the user.

NF1.2 The system will not transmit any data which was necessary to collect or generate, including log files, without obtaining explicit consent from the user.

2 Software Quality

The following list specifies the system's core requirements in terms of portability, source control, testability, usability and documentation. There are no specific performance metric requirements for the system at this stage of its development.

NF2.1 The system must not use any platform-specific libraries, functions or commands.

NF2.2 The system must provide a `requirements.txt`[5] file or similar, to allow its dependencies to be installed using Pip.

NF2.3 The system must be versioned and privately hosted on GitHub.

NF2.4 The implementation of the system should include a severity-based logging facility which writes to a text file, for use during debugging and testing.

NF2.5 The system must provide a non-interactive help facility for users.

NF2.6 The system should provide visual feedback during computationally expensive tasks to show task progress, e.g. with loading bars.

[5] https://pip.readthedocs.io/en/1.1/requirements.html

Appendix B

Implementation Raw Data

B.1 Token/Entity Data

Table B.1: Entities as a percentage of total tokens from 40 BBC Politics articles.

Tokens	Entities	Entities (%)
81	7	8.64198
129	8	6.20155
167	9	5.38922
181	15	8.28729
252	17	6.74603
264	15	5.68182
272	11	4.04412
275	17	6.18182
291	21	7.21649
343	22	6.41399
349	14	4.01146
366	23	6.28415
367	28	7.62943
367	14	3.81471
404	42	10.39604
409	40	9.77995
413	41	9.92736
434	25	5.76037
444	20	4.5045
453	29	6.40177
480	28	5.83333
498	41	8.23293
503	30	5.96421
523	35	6.69216
527	17	3.22581
536	62	11.56716
562	39	6.9395
562	32	5.69395

Table B.1: Entities as a percentage of total tokens from 40 BBC Politics articles.

Tokens	Entities	Entities (%)
571	42	7.35552
618	33	5.33981
621	38	6.11916
674	35	5.19288
688	34	4.94186
710	63	8.87324
722	65	9.00277
831	76	9.14561
920	51	5.54348
987	83	8.40932
1119	52	4.64701
1243	52	4.18343