Automatic Semantic Analysis for Academic Writing

Support

Stephen T. O'Rourke
Master of Philosophy
School of Electrical and Information Engineering,
University of Sydney

2009
Abstract

This thesis evaluates methods for analysing the semantic structure and flow of an essay and presents a novel visualisation method for providing feedback to support formative assessment in academic writing. The visualisation makes use of text mining techniques to provide insight on the semantics of the topics in an essay. This thesis argues that automatic techniques for semantic analysis and visualisation can be used to mitigate many of the problems associated with the subjectivity of essay assessment by allowing greater insight into an essay’s latent features. Many approaches have been proposed for providing feedback in academic writing; however, few of them are visually-based. The research presented here is a development of several strands of previous research in text mining and visualisation and fits within a larger effort to provide feedback to support academic writing. The proposed essay visualisation method involves a process of Non-negative Matrix Factorisation to uncover topics in an essay, followed by Multidimensional Scaling to map the topic closeness of the essay’s paragraphs. The visualisation method is evaluated with a three corpora of short essays written by university students.
Acknowledgements

I would like to express my sincerest thanks and appreciation to my supervisor, Dr. Rafael Calvo, for his attention, patience and guidance during the course of this research and writing of this thesis.

A special thanks to Jorge Villalon for his friendship and constructive discussions in the development and evaluation of the research ideas presented in this thesis.

In addition, I would also like to recognise that this research was financially supported in part by an Australian Research Council scholarship, for which I am extremely grateful.
Preface

During the course of this research, a number of papers have been published based on the work presented in this thesis. These papers are listed here and included as appendices at the end of the thesis for reference.


Contents

Chapter 1: Introduction ..............................................................................................................1

Chapter 2: Analysing the Semantics of Texts .................................................................8
  2.1 Linguistic Approaches .....................................................................................................8
  2.2 Text Mining Approaches .............................................................................................13
    2.2.1 Text Pre-processing ...............................................................................................14
    2.2.2 Vector Space Model ..............................................................................................15
  2.3 Topic Models ................................................................................................................19
    2.3.1 Latent Semantic Analysis ......................................................................................19
    2.3.2 Non-negative Matrix Factorisation .......................................................................22
    2.3.3 Probabilistic Topic Models ....................................................................................23
    2.3.4 Understanding the Topic Semantic Space ............................................................26

Chapter 3: Visualising the Semantics of Texts ...............................................................30
  3.1 Proposition-based Maps ..............................................................................................31
  3.2 Scatterplot-based Maps ..............................................................................................34
  3.3 Proximity-based Maps ................................................................................................39
    3.3.1 Multidimensional Scaling ....................................................................................41
    3.3.2 Classic Multidimensional Scaling .........................................................................43
    3.3.3 Metric Multidimensional Scaling ..........................................................................44
    3.3.4 Non-metric Multidimensional Scaling ..................................................................45

Chapter 4: Glesser - Automatic Semantic Feedback Tool .............................................48
  4.1 Design and Architecture ..............................................................................................50
  4.2 Feedback Tools ..............................................................................................................55

Chapter 5: Mapping the Semantics of Texts .................................................................64
  5.1 Data Analysis .................................................................................................................64
  5.2 Quantifying Topic Flow ...............................................................................................67
  5.3 Topic Flow Map ............................................................................................................68

Chapter 6: Evaluation: Data and Methods .................................................................73
  6.1 Experiment Datasets ....................................................................................................73

iv
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.2</td>
<td>Experiment 1: Quantifying Topic Flow</td>
<td>75</td>
</tr>
<tr>
<td>6.3</td>
<td>Experiment 2: Visualising Topic Flow</td>
<td>81</td>
</tr>
<tr>
<td>6.4</td>
<td>Experiment 3: Measuring Visualisation Insight</td>
<td>85</td>
</tr>
<tr>
<td>6.5</td>
<td>Discussion and Future Work</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>Chapter 7: Conclusions</td>
<td>95</td>
</tr>
<tr>
<td></td>
<td>References</td>
<td>97</td>
</tr>
<tr>
<td>Appendix A</td>
<td>MASUS Diagnostic Assessment Sheet</td>
<td>102</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Analysing Semantic Flow in Academic Writing</td>
<td>103</td>
</tr>
<tr>
<td>Appendix C</td>
<td>Visualizing Paragraph Closeness for Academic Writing Support</td>
<td>112</td>
</tr>
<tr>
<td>Appendix D</td>
<td>Visualising Social Networks in Collaborative Environments</td>
<td>118</td>
</tr>
</tbody>
</table>
Tables

Table 3.1: A guideline of quality judgements for MDS configurations proposed by Kruskal (1964)........................................................................................................43
Table 6.1: The mean sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods on the BAWE corpus..............78
Table 6.2: The mean sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods on the MSU corpus.................78
Table 6.3: The mean sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods on the MASUS corpus. ........80
Table 6.4: A comparison of the statistical differences between the MASUS essay subset with maps and the MASUS essay subset with no maps.........................88
Table 6.5: A summary of time taken for the two raters to score the MASUS essays, both with and without the visual aid of the topic flow maps. .................................89
Table 6.6: A summary of the scores given by the two raters and their correlation with the expert MASUS rater.................................................................91
Figures

Figure 2.1: A prototypical example adapted from Grosz et al. (1995), demonstrating sentence to sentence coherence. Discourse 1 reads more coherently than Discourse 2 despite both discourses containing the same information. ..........10

Figure 2.2: An example tree diagram of an RST analysis. Figure reproduced from Taboada and Mann (2006). ................................................................. 12

Figure 2.3: An illustration of the geometric difference between Euclidean distance (dotted line) and the cosine of the angle \( \theta \) between two vectors, \( d_1 \) and \( d_2 \). ..........18

Figure 2.4: Plate notation representing the PLSA (Left) and LDA (Right) topic models. The figure has been adapted from Blei et al. (2003). .................25

Figure 3.1: A concept map describing the keys features of concept maps. Figure reproduced from Novak (2006). ................................................................. 32

Figure 3.2: PNAS articles coloured according to their biology subfield categories. The two-dimensional view on the three-dimensional space was selected algorithmically (Left) and by aided human selection (Right). Figure reproduced from Landauer et al. (2004) ................................................................. 36

Figure 3.3: A scatterplot of the conceptual coherence of team participants. Figure reproduced from Vande et al. (2008). ................................................................. 38

Figure 4.1: The user interface of the Topics tool in Glosser. The trigger questions are displayed at the top of the page and ‘gloss’ below ...........................................49

Figure 4.2: A class diagram outlining the structure of Glosser’s extendible architecture. ................................................................. 52

Figure 4.3: A data flow diagram describing the process of harvesting and indexing a document in Glosser ................................................................. 54

Figure 4.4: A Java class exemplifying how to create simple spell-checking tool in Glosser ................................................................. 56

Figure 4.5: An example of the ‘gloss’ output produced using the Readability Slider tool. A warning icon is displayed when the difference in writing style between two paragraphs is below the threshold value of the slider ...........................................57
Figure 4.6: An example of the ‘gloss’ output produced using the Coherence Slider tool. A warning icon is displayed when the topic overlap between two paragraphs is below the threshold value of the slider. ........................................58

Figure 4.7: A topic flow map consisting of five paragraphs. The similarity of the paragraphs’ topic content is represented by their relative proximity in the map. ........................................................................................................59

Figure 4.8: A topic coverage map generated from a short essay. The colour of the nodes and arrows indicates the topic of the associated keywords and their connecting sentences. .....................................................................................61

Figure 5.1: A map of the topic flow generated from the paragraphs of a short essay.70

Figure 5.2: A map of an essay with a high grade (Left) and an essay with a low grade (Right). Note the low grade essay has many Rough-Shifts in topics while the high grade essay has a more gradual topic flow that starts and finishes on a similar point ........................................................................................................71

Figure 6.1: A comparison the average sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods for the BAWE corpus. ........................................................................................................77

Figure 6.2: The mean sentence distance index (DI) for the exact numerical grades in the MSU corpus........................................................................................................................................79

Figure 6.3: A scatterplot of the fit error in an MDS configuration versus the number of paragraphs in each document for semantic spaces with different k dimension constrains. .............................................................................................83

Figure 6.4: A scatterplot of the Pearson’s correlation between then distances in the NMF model and the distance in an MDS configuration versus the number of paragraphs in a document for semantic spaces with different k dimension constraints. ..........................................................................................84

Figure 6.5: A diagram illustrating the essay assessment process with the visual aid of an essay’s topic flow map.........................................................................................................................86

Figure 6.6: The main interface of the essay assessment tool. An essay (Left) is displayed side-by-side with it corresponding map (Right). .................................................................87
Chapter 1

Introduction

In academic settings, an essay is a standard assessment task, considered by many educators to be one of the most useful tools for assessing one’s writing ability and attainment of learning outcomes. While learning outcomes imply the ability to recall, organise and relate ideas to satisfy some objective, it is the ability to express and structure one’s answer through writing that is essential to communicate this knowledge effectively. Thus, a common challenge for many learners in constructing a quality essay answer is the skill of writing itself, and more specifically, producing a coherent argument.

Formative assessment refers to the process of helping a learner to improve their performance by providing them with feedback on the quality of their work (Sadler, 1989). Sadler argues that for a learner to best benefit from feedback they need to be supported in understanding i) what good performance is, ii) how current performance relates to good performance, and iii) how to act to close the gap between current and good performance. In effect, this makes formative assessment a cognitive process, which requires the learner to be actively engaged in interpreting, constructing and internalising the feedback to bring about any change in performance (Nicol & Macfarlane-Dick, 2006). The process of understanding feedback, however, can often be complicated by misconceptions and confusion between the learner’s view of their work and that of the feedback provided (Ivanic, Clark, & Rimmershaw, 2000). Thus, the way in which feedback is communicated is vital to helping learners to actively develop and change their understanding. Importantly, feedback also needs to be timely in order to ensure relevance and allow for adjustment on the learner’s part (Freeman & Lewis, 1998). These issues provide
an indication of the need for highly focused approaches in delivering feedback in formative assessment.

Essay assessment procedures are designed to produce accurate and consistent numerical scores of individual essays from multiple assessors. These assumptions place enormous reliance on the technology of grading, such as scoring guidelines and the statistical calculation of inter-assessor agreement. These technologies seek to codify assessment standards and bring greater accuracy and objectivity to the assessment process. It is through these grading technologies that one can isolate, assess and provide feedback on the different aspects of learners’ writing.

Essay assessment, however, is a highly subjective process that is prone to several types of errors (Rudner, 1992), leading to many inconsistencies in the grades given by different assessors. Similarly, assessing to provide a grade or feedback is enormously time-consuming, and thus, a costly process. While some aspects of an essay can be assessed based on objective features, such as the word count or spelling errors, others require a more subjective interpretation, such as the flow of topics and ideas. These subjective aspects of essay assessment differ significantly in that they are somewhat inexact and cannot be easily abstracted without an in-depth analysis.

Some researchers have sought to address the complexities of essay assessment using computational methods to reduce the costs and to increase objectivity. Since the emergence of computer-assisted essay assessment in the 1960s, many lines of research have emerged to support the various stages of the writing process. This research has resulted in the developed of a number of software tools to support the essay writing process by providing Automated Essay Scoring (AES) to grade essays (Burstein, 2003; T. K. Landauer, Laham, & Foltz, 2000; Page, 2003) as well as automated feedback to identify problems (Britt, Peter, Aaron, & Charles, 2004; Macdonald, Frase, Gingrich, & Keenan, 1982) and highlight important features for reflection (Villalon, Kearney, Calvo, & Reimann, 2008).

The adoption of AES is one approach that has been proposed to bring greater consistency and objectivity to the essay assessment process. The assumption made by most AES researchers is that the scores given by human assessors represent the most
accurate measure of an essay’s quality. Thus, the aim of most AES systems is to simulate the human assessment process as accurately as possible. AES systems typically use essays manually scored by humans to build an essay-specific assessment model, which can then be used to approximate the expected grades for unscored essays automatically.

While AES has been shown to have inter-assessor correlations comparable to that of human assessors (Dikli, 2006), many scholars are still highly critical of the validity and robustness of the approach. Some scholars suspect that the reportedly high inter-assessor agreement between AES and human assessors might be indirectly due to elements that commonly occur in well written essays, such as the use of a rich vocabulary (Calfee, 2000), rather than a direct assessment of the essay answer. Critics of AES are also concerned about the effects of students ‘unintelligently’ writing to the audience of the machine, and attempting to match its ideal ‘formulaic’ model at the expense of logical argument (Cheville, 2004). Overall, many scholars remain unconvinced that AES systems can simulate the complexities needed to accurately score an essay. These arguments against AES largely stem from the fact that essay assessment is subjective, making a purely objective assessment approach contradictory.

Due to the associated problems with AES, along with solid pedagogical reasoning, many researchers have focused on going beyond the automated scoring of an essay to provide the writer with automated feedback instead. Automated feedback tools typically use the same techniques as AES to extract essay features, but instead attempt to meaningfully translate this data into useful information. The feedback generated helps the writer to improve their performance based on the relationship between specific essay features and common writing problems. One of the challenges of this approach is to generate feedback that is specific enough to help the writer understand why a particular feature is assessed the way it is.

Many automated essay feedback tools have been developed by researchers that are specifically targeted towards common problems in writing. The Writer’s Workbench tool provides feedback on spelling, style and diction by analysing
English prose and suggesting possible improvements (Macdonald et al., 1982). The Source's Apprentice Intelligent Feedback tool (Britt et al., 2004) provides automatic feedback on sourcing by detecting citations and plagiarised sentences and suggesting ways to resolve them. More recently, the Glosser tool (Villalon et al., 2008) seeks to provide automatic feedback by highlighting important essay features and using a generic set of thought provoking questions to trigger reflection. This approach of triggering reflection in writing is supported by Reynolds and Bonk (1996), who showed that even the use of a simplistic set of generic messages could be used to promote revision in a writing activity. The results of the study showed that the group of students who were given message prompts performed more revisions that were meaningful and produced better quality texts than those who received no message prompts.

Research has shown that certain semantic features are more evident in higher graded essays. In particular, a strong correlation has been found between topic continuity and essay grades (Rogers, 2004). The existence of this relationship is also theoretically supported by the MASUS procedure (Bonanno, Jones, & University of Sydney. Learning Centre., 2007), which lists semantic features, such as a ‘clear flow of ideas’ and a ‘conclusion that follows from the rest of the essay’, in its criteria. These semantic features are related to the cohesion of an essay, which is definable as the continuality of semantic elements among successive parts of a text. A lack of cohesion is a common problem in essay writing, which can be most evidently seen in transitions among sentences and paragraphs, and, to a lesser degree, sections and chapters. While a lack of cohesion can at times provide meaning in itself, nevertheless, knowing to what extent a gap exists and understanding how it semantically relates to the rest of the text is not always intuitive.

Fahnestock (1983) identifies two kinds of cohesive transitions in writing, semantic and lexical, which translate to the ability of the writer to communicate a coherent argument. Coherence explains how transitions between a text’s parts combine to give it structure and meaning as a whole, as opposed to cohesion, which explains how one part transitions to the next. In this sense, transitions between parts can be thought of as relating to the ‘flow’ of the text. Lexical transitions operate on
the surface structure of a text, while semantic transitions reveal deeper meaning in relations between successive ideas or subtopics of the text. The primary focus of this thesis is on the latter, and more specifically, on the semantics of a text’s topic flow.

Evaluating topic flow, however, is a highly subjective task that does not always have a definitive answer. While quantitative measures, such as cohesion, can be useful for arriving at an answer, they generally offer more of a guide to a text’s connectedness, which does not necessarily correlate well to that of experts. Topic flow is subjective, and thus, needs to be assessed in a subjective way. One novel and largely unexplored area for assessing the semantic aspects of essays is the field of information visualisation. Information visualisation is essentially the study of how data can be represented visually for maximum cognition. Visualisation offers a novel approach for analysing a qualitative dataset, which can be specifically adapted to enhance the speed and accuracy of a subjective task. In this way, visualisation can be used to emphasise existing or latent features of a dataset, which may otherwise be cognitively difficult to uncover in a non-visual way. Hence, in this approach the user, rather than an algorithm, plays a central role in interpreting the dataset. While a full discussion of how the cognitive aspects of visualisation relate to human visual perception is not in the scope of this thesis, a thorough analysis of this line of research is presented by Ware (2000).

In the information visualisation literature, the quality of cognitively discovering that which is characteristically complex, deep, qualitative, unexpected and relevant in a dataset is termed ‘insight’. North (2006) discusses how features possessing the characteristics of insight can be abstracted from complex datasets, such as that of unstructured text or a high-dimensional vector model, by transforming the data to a low-dimensional visual representation. This approach provides a qualitative view of a dataset, which has been shown to help increase user efficiency when completing subjective tasks, such as the characterisation of latent structures and relationships (Fayyad, Grinstein, & Wierse, 2002), by allowing humans to rapidly access large amounts of data. A great deal of research has been conducted in using various visualisations to enhance the analysis of qualitative datasets. Visualisation has previously been applied to many educational datasets to bring greater insight to
learning tasks, such as the awareness of participation in collaborative groups (Kay, Maisonuneuve, Yacef, & Reimann, 2006) and social interaction in collaborative environments (O’Rourke & Calvo, 2009b).

The hypothesis of this thesis is that visualisation can be used to mitigate many of the problems associated with the subjectivity of essay assessment by bringing greater insight to an essay’s latent features. In this approach, an essay visualisation contains exactly the same information as its equivalent text, but uses visual techniques to emphasise specific features relevant to essay assessment. Using the rubric of the MASUS procedure, this thesis will identify the semantic features most relevant to essay assessment. This visualisation approach differs from previous computer-assisted assessment approach in that it does not attempt to automatically score or highlight certain features for reflection, but rather uses these features to present the same information in a new and meaningful way. Thus, the visualisation approach offers an insight into an essay’s latent features while still maintaining the subjectivity of the assessment process.

This thesis examines the applicability of document similarity comparisons for quantifying the semantic features of an essay and how visualisation can be used to interpret these features. This thesis does not, however, contribute to evaluating how such techniques can be best applied for the purposes of linguistics or teaching and learning, although such applications are mentioned and background is given to provide a context for this research. Rather, the focus of this research is on the technical aspects, and on how automated techniques can be used, firstly to quantify, and secondly to interpret this information. The broader application of such techniques is left to researchers in their respective fields.

The next chapter, Chapter 2, reviews some of the currently available automated assessment methods and the extensive research for analysing the semantics of academic writing. This is followed by an introduction to the larger research context of formative assessment and a comparison with previous work to highlight the novelty of the contribution of this thesis. Chapter 3 presents an overview of algorithms and techniques, which can be used to visualise the semantic features of a
text. Chapter 4 presents the automatic semantic feedback tool, called Glosser, a collaborative effort to which this thesis is contributing. In Chapter 5, the approach for quantifying topic flow is introduced and a visualisation for interpreting essay semantics is introduced and explained using examples. Chapter 6 provides an evaluation of the algorithms used for measuring the essay semantics and assesses to what extent the essay visualisation allows one to capture semantic features relevant to the MASUS procedure. Finally, Chapter 7 concludes the thesis.
Chapter 2

Analysing the Semantics of Texts

This chapter presents an overview of some of the many linguistic and text mining techniques for analysing the semantics of texts, with particular reference to the current state of Automated Essay Scoring (AES) research. Further, it discusses how these techniques can be used to enable automatic feedback and provide a reader with an insight into an essay’s latent semantic features.

2.1 Linguistic Approaches

The semantics of a text describe the relationships between the linguistic elements that connect its sentences, paragraphs and sections. The semantic flow of these elements translates to how these connections transition to create a coherent argument. A lack of semantic flow can be a problem when meaning becomes lost from one part of the text to the next.

Many quantitative measures have been developed for analysing the semantics of writing, with different applications for representing meaning, comprehension and content. These measures can be used to identify potential problems in the semantic flow among the elements of a text based on various features of a text, such as characteristics of its words (Flesch, 1948), cohesive devices (Halliday & Hasan, 1976), sequences of propositions (Kintsch & Vandijk, 1978), sentence transitions (Grosz, Weinstein, & Joshi, 1995), and the intention of its parts (Taboada & Mann, 2006). For example, a rough-shift in the semantics between two paragraphs could occur when topics are short-lived and unconnected, and thus be indicative of poor topic
development. This section presents a review of some of the many methods proposed by linguistics for analysing the different semantic features of texts.

**Readability Tests**

An early attempt to analyse text semantics were readability tests, which were first introduced by Flesch (1948). Readability tests attempt to score the complexity of a text based on the length of sentences and the characteristics of its words. The Flesch Reading Ease Score (FRES), described in Equation (2.1), gives a 0–100 score that translates to the comprehension difficulty of text. A higher FRES indicates text that is easier to comprehend while a lower FRES indicates the opposite. Although many other readability tests have been proposed, they all typically use the same core measures of sentences, words and syllables, which are weighted by different factors.

\[
FRES = 206.835 - 1.015 \left( \frac{\text{total words}}{\text{total sentences}} \right) - 84.6 \left( \frac{\text{total syllables}}{\text{total words}} \right)
\]  

(2.1)

Readability tests have also been applied for plagiarism detection (Torres & Roig, 2005). Plagiarism is a known problem in academic writing and comes in many different forms (Lukashenko, Graudina, & Grundspenkis, 2007), some of which can be identified through rough-shifts in the semantic flow of a text’s writing style. Even through these problems do not provide proof of plagiarism, it is clear that these are undesirable features in academic writing for which readability measures could be used as a metric to identify problems.

**Text Cohesion**

Cohesive relationships joining the parts of a text have definable purposes that facilitate a writer in creating meaning. Text cohesion is a method for analysing and coding the semantic relations between sentences to determine how connected a text is. The approach is based on the grammatical features of the text. In their definitive paper on text cohesion, Halliday and Hasan (1976) provide a detailed explanation
and examples of the different types of these cohesive relationships; namely, reference, substitution, ellipsis, and conjunction.

**Centring Theory**

Centring Theory (CT) is a model for measuring the flow of coherence in sentence transitions (Grosz et al., 1995). The development of CT was largely motivated by the need to formalise a notion of connectedness in text in order to explain why, for example, in Figure 2.1 the sentences in Discourse 1 are more coherent than the sentences in Discourse 2 despite both examples containing the same information. CT objectifies this difference by explaining that in Discourse 1, the ‘centre’ topic of John is being maintained from sentence to sentence, while in Discourse 2, the ‘centre’ topic switches back and forth between John and the store.

**Discourse 1.**
- a. John went to his favourite music store to buy a piano.
- b. He had frequented the store for many years.
- c. He was excited that he could finally buy a piano.
- d. He arrived just as the store was closing for the day.

**Discourse 2.**
- a. John went to his favourite music store to buy a piano.
- b. It was a store John had frequented for many years.
- c. He was excited that he could finally buy a piano.
- d. It was closing just as John arrived.

**Figure 2.1:** A prototypical example adapted from Grosz et al. (1995), demonstrating sentence to sentence coherence. Discourse 1 reads more coherently than Discourse 2 despite both discourses containing the same information.

The idea of CT is that there are numerous topics at the centre of attention in a text at any one point and that the coherence of a text is affected by how these topics transition from one part to the next. CT defines two types of ‘centres’, an ordered set of forward-looking centres, $C_i$, which provide links to upcoming parts, and a single backward-looking centre, $C_b$, which links back to the previous part. $C_b$ can roughly
be considered to be the current topic in the text, while $C_t$ can roughly be considered to be the next topic in the text.

In a CT analysis, the topics of $C_t$ are ranked in order of importance, with the most highly ranked topic being the preferred centre. When $C_b$ is the most highly ranked topic of $C_t$, this indicates that it will continue to be the topic in the next part of the text. This situation is known in CT as a Continue Transition. When some entity other than $C_b$ is the most highly ranked topic of $C_t$, an upcoming topic shift is signalled. This situation is known in CT as a Retain. A topic shift following a Retain will be a Smooth-Shift. If the topic shifts in such a way that it is neither the preferred centre of the current part nor of the previous part, then the shift will be a Rough-Shift.

Miltakaki and Kukich (2004) applied CT to AES by using a Rough-Shift-based metric to capture incoherence. Their results indicated the AES system improved its performance significantly, better approximating human scores while also providing the capability of providing feedback to the student.

**Rhetorical Structure Theory**

Rhetorical Structure Theory (RST) attempts to model the structure of a text by analysing the intention of it parts (Taboada & Mann, 2006). RST views each part of the text as having a function that relates to the overall text structure. However, unlike CT, RST does not consider topic relations; rather, it uses rhetorical relations to define the function of a particular part of the text. Thus, RST provides a measure of coherence that is independent of other measures.

In their definitive paper on RST, Mann and Thompson (1988) define a set of 23 common rhetorical relations to link parts of texts, such as Elaboration, Condition, Means, Preparation, Purpose and Summary. These rhetorical relations provide a basis for the various types of relationships within the text. A text can be represented in an RST analysis as a rhetorical structure tree with a unique root that spans the entire text. Each element in the tree has a specific role that links to some other
element. Breaks in this rhetorical structure can indicate incoherence in a text. Figure 2.2 illustrates an example rhetorical structure tree from a coherent sentence.

Figure 2.2: An example tree diagram of an RST analysis. Figure reproduced from Taboada and Mann (2006).

In summary, the methods presented thus far provide an overview of some of the various linguistic techniques for analysing semantics in writing. Readability tests are specifically directed towards analysing the complexity of language and are an amenable method for analysing global coherence in a general sense, but do not directly consider the actual topic content. CT provides a useful model for measuring the topic coherence between the consecutive parts of a text, but does not analyse the text globally as a whole. RST addresses coherence by stipulating that adjacent parts of a text be connected by rhetorical relations, but does not consider the coherence of the text’s actual topic content.
2.2 Text Mining Approaches

Text mining refers to the process of extracting information from unstructured texts. It is an interdisciplinary field that draws on techniques from information retrieval, data mining, machine learning, statistics and natural language processing. More recently, text mining-based approaches have been applied to analyse the semantics of texts. This typically involves performing computations on a text’s parts to uncover latent information that is not directly visible in the classical format available on a text’s surface.

A text mining analysis involves several challenging problems, mainly influenced by the fact that texts, from a computational perspective, are simply unstructured collections of words. The standard theoretical framework for text mining typically involves the steps of pre-processing to structure the input text, extraction of the features from the structured data to create a model, and finally, evaluation and interpretation of the model output.

Text mining has been applied to solve a number of problems in educational settings, including automatic summarisation, automatic assessment, automatic tutoring and plagiarism detection (Dessus, 2009), as well as in combination with other text-associated information, such as the social dimension of the texts’ authors (O’Rourke & Calvo, 2008). These applications use techniques based on the semantics of the document similarity comparisons. The most common approach involves the creation of a term-by-document matrix, derived from frequency vectors of distinct terms in each document to create a topic model of a document corpus.

The most representative topic modelling techniques in the research used in conjunction with a term-by-document are Latent Semantic Analysis (LSA) (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990), Non-negative Matrix Factorisation (NMF) (Lee & Seung, 1999), Probabilistic LSA (PLSA) (Hofmann, 2001) and Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003). These are known as dimensionality reduction techniques, where individual terms in the term-by-document matrix are weighted according to their significance in the topic. The aim of
the dimensionality reduction is to eliminate unimportant details in the data and to allow the latent underlying semantic structure to become evident.

Performing a text mining analysis requires the choice between a number of algorithmic techniques and parameters, which are determined based on the application domain, theoretical grounds, or empirical evidence. In this section, text mining techniques for analysing the semantics of texts are detailed and the decisions needed to provide an application for the specific purposes of automatic essay feedback are discussed.

### 2.2.1 Text Pre-processing

In order to construct the term-by-document matrix, pre-processing is first required to cleanup and to extract a structured representation from the raw text. This is a core component in text mining research and includes techniques, such as tokenisation, stemming and stopword removal, to structure the data for further analysis. This section briefly presents these typical pre-processing steps.

**Stopword Removal**

One of the first pre-processing steps undertaken by text mining systems is to identify terms for analysis. In order to avoid analysing irrelevant terms that may create ‘noise’ (Van Rijsbergen, 1979), a list of stopwords is commonly associated with a document corpus. Stopwords are words that are so frequent in a language that their informational contribution to a topic is almost nothing. For example, words such as ‘is’, ‘a’ and ‘the’ are usually counted as stopwords. The two most commonly used English language stopword lists in the literature are the Van Rijsbergen and Brown stopword lists, which contain 319 and 425 words respectively (Fox, 1992).
Stemming

Stemming is a technique used to group similar words, which are small syntactic variations of one another, by removing word suffixes to retrieve their root ‘stems’. It is a common technique used in text mining, as it reduces complexity with little informational loss in the context of a topic. For example, the words ‘assess’, ‘assessing’ and ‘assessed’ all carry the same root stem, ‘assess’, and can thus all be viewed as different occurrences of the same word. One of the most commonly used stemming algorithms for the English language is the Porter algorithm (Porter, 1980), which strips such morphological and inflectional word suffixes.

2.2.2 Vector Space Model

The Vector Space Model (VSM) is a ‘bag-of-words’ modelling approach, which represents text documents as vectors in a liner space (Salton, Wong, & Yang, 1975); where a document could be any finer-grained sentence, paragraph or text segment within a text. In Salton’s (1975) classic VSM model, each distinct term in a document corpus corresponds to a single dimension in a vector space. This vector space can be mathematically represented as a term-by-document matrix $X$, described in Equation (2.2), defining the semantic relationships between documents through their corresponding term occurrences. An element $x_{ij}$ of $X$ indicates the weight (usually derived from a term weighting scheme) of a term $t_i$ in a document $d_j$. The term weights can be viewed as the features describing the document.

$$
X = \begin{bmatrix}
  x_{11} & \cdots & x_{1n} \\
  \vdots & \ddots & \vdots \\
  x_{m1} & \cdots & x_{mn}
\end{bmatrix}
$$

(2.2)

The VSM model has the advantage of being able to model documents and terms at both the local level, in a single document, and at the global level, in the entire document corpus. The basic premise of the VSM model is that documents which are
written about similar topics will have more words in common than the documents which are written about dissimilar topics. In this way, the VSM model can be used to perform operations based on the similarity of documents’ topic mixture. For example, by computing the semantic similarity or ‘distance’ between pairs of documents, documents with similar topic content can be identified.

Variations of the VSM model have been applied to solve a number of problems, such as document clustering (Bao, Tang, Li, Zhang, & Ye, 2008), automated essay scoring (Kakkonen, Myller, Sutinen, & Timonen, 2008), information retrieval (Raghavan & Wong, 1986) and automatic indexing (Salton et al., 1975). These methods extract meaning from texts by applying some sort of dimensional reduction technique over the document vector representation. Instead of terms, the resulting models represent documents as a set of feature dimensions, which can be thought of as representing the main topics the documents.

**Term Weight Schemes**

Since terms tend to vary in the informational value they contribute to a topic in the context of a document and its larger document corpus. Term weighting schemes are typically used to normalise any biases caused by this effect. Generally, the weighting of a term increases proportionally to its local frequency in a document, offset by its global frequency in the document corpus. As such higher weights are assigned to the ‘important’ terms, which occur fewer times in a small number of documents, eliminating the noise of more common terms.

Information retrieval and AES are two applications where term weighting schemes have been shown to enhance performance in most cases. This can be explained by the enrichment of the document semantic model to favour the more important terms, which can better represent a document’s topic mixture. Several term weighting scheme choices that are commonly considered in the literature when constructing a weighted term-by-document matrix are Log-entropy and TD-IDF (Ricardo & Berthier, 1999).
The formula for calculating the TF-IDF weight of a term $t_i$ for a document $d_j$ in a corpus $D$ is mathematically described in Equation (2.3). Where the term frequency (TF) is the number of times a term occurs in a document divided by the total number of terms in the document. The inverse document frequency (IDF) is the logarithm of inverse of the number of documents in which the term occurs $d : t_i \in d$ divided by the number of documents in the corpus $|D|$. This TF-IDF formula assigns weights between 0 and 1 depending on a term’s frequency distribution.

$$x_{ij} = tf \times idf$$

$$= \frac{f_{ij}}{\sum_k^n f_{kj}} \times \log \left( \frac{|D|}{d : t_i \in d} \right) \quad (2.3)$$

If, however, the documents to be analysed have some sort of topical categorisation that contains relevant information, the words can also be weighted according to their Shannon entropy over the document corpus. Log-entropy weighs a term by the log of its local frequency $f_i$ in a document offset by the inverse of the Shannon entropy of its global frequency across all $n$ documents in the corpus. The formula for weighting a term using Log-entropy is mathematically described in Equation (2.4). This formula assigns weights between 0 and 1 depending on a term’s frequency distribution.

$$x_{ij} = \frac{\log(1 + f_{ij})}{-\sum_k^n \left( \frac{f_{ik}}{\sum_l^n f_{il}} \right) \log \left( \frac{f_{ik}}{\sum_l^n f_{il}} \right)} \quad (2.4)$$

**Distance Measures**

Distance measures are used to compute the semantic relatedness between document vectors in a VSM model. Numerous distance measures have been proposed in the literature for measuring different types of vector models and
attributes. Understanding the differences and relationships among these distance measures is essential to tailoring a solution to a particular application. This section compares two commonly used distance measures, namely, Euclidean distance and Cosine similarity, which are based on the geometric distance and the angle between vectors, respectively.

![Figure 2.3: An illustration of the geometric difference between Euclidean distance (dotted line) and the cosine of the angle \( \theta \) between two vectors, \( d_1 \) and \( d_2 \).](image)

Figure 2.3 illustrate two vectors in a two-dimensional space. If a straight line is drawn from \( d_1 \) to \( d_2 \), we can define the geometric distance between the two points. This is the called Euclidean distance. The Euclidean distance between documents \( d_i \) and \( d_j \) in an n-dimensional Euclidean space is mathematically expressed in Equation (2.5).

\[
\begin{align*}
d(i, j) &= |d_i - d_j| \\
&= \sqrt{(x_{1i} - x_{1j})^2 + (x_{2i} - x_{2j})^2 + \ldots + (x_{ni} - x_{nj})^2}
\end{align*}
\]  

(2.5)

Cosine similarity is a distance measure for ranking and comparing documents commonly used in information retrieval, whereby documents are ranked according to how similar they are to a query. While the definition of ‘similar’ used is subjective, it has been shown to be highly similar to that of humans (T.K. Landauer, McNamara, Dennis, & Kintsch, 2007). Figure 2.3 provides an example of the angle \( \theta \) between two document vectors. As the angle between a pair of document vectors decreases, the
similarity in the semantics of their content increases (i.e., the cosine of the angle approaches 1). Mathematically, cosine similarity is defined as the normalised dot product of two vectors, giving values in the range of -1 and 1, or 0 and 1, when all attributes of document vectors are non-negative. The formula for calculating cosine similarity in an n-dimensional space is expressed in Equation (2.6).

\[
\cos \theta(i, j) = \frac{\mathbf{d}_i \cdot \mathbf{d}_j}{|\mathbf{d}_i| \times |\mathbf{d}_j|}
\]

\[
= \frac{\sum_{k} x_{ki} x_{kj}}{\sqrt{\sum_{k} x_{ki}^2 \times \sum_{k} x_{kj}^2}}
\]

(2.6)

### 2.3 Topic Models

There is no single topic modelling method that can be suitably applied to all problems. Thus, a large number of topic modelling approaches have been developed for different applications. This section presents a number of dimension reduction algorithms, which can be used to infer topic models from texts.

#### 2.3.1 Latent Semantic Analysis

Latent Semantic Analysis (LSA) extends the basic premise of the VSM model to help infer the underlying latent relations among terms, which are not directly apparent from an analysis of terms co-occurrences alone (Deerwester et al., 1990). Like the VSM model, a document in the LSA model is represented as a sum of its comprising term vectors, however, the dimensions of these term vectors fortuitously model the topics and concepts in the document corpus.
LSA is an unsupervised method for learning a topic model. It uses a matrix factorisation technique called Singular Value Decomposition (SVD) to decompose a low rank approximation of a term-by-document matrix. This low rank approximation identifies a set of basis vectors that capture most of the variance of the corpus in a linear space. These basis vectors can be linearly combined to represent any document in the space, thus allowing documents to be indirectly related through the semantics of the basis vectors they span.

The LSA algorithm uses SVD to linearly transform the term-by-document matrix to a reduced vector space. SVD performs a $k$ rank decomposition of $X$, where $k$ is the number of latent topics, $U$ is a topic-by-term matrix, $V$ is a matrix of document-by-topic vectors, and $\Sigma$ is a diagonal matrix of singular values, such that $X = U \Sigma V^T$. This formula is mathematically expressed in Equation (2.7).

$$X = (t_i^T) \rightarrow \begin{bmatrix} u_1 \ldots u_l \end{bmatrix} \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_l \end{bmatrix} \begin{bmatrix} v_1 \\ \vdots \\ v_l \end{bmatrix}$$ (2.7)

Foltz (1998) used LSA to measure the coherence of a document by calculating the degree of semantic relatedness between consecutive text passages. In addition to this, Foltz also used a sliding window approach to measure coherence on a more global level, by comparing the semantic variability between groups paragraph. The advantage of the sliding window is that it can smooth the effects of very local coherence changes, which can occur from paragraph to paragraph, and instead focus on the general semantic change in topic over a text. Using LSA, Foltz was able to predict the effect of text coherence on comprehension successfully. These results were based on the analyses of artificially created texts and comprehension measures taken from two previous studies by Britton and Gulgoz (1991) and McNamara et al. (1996).
In the first set of texts, from Britton and Gulgoz (1991), the text coherence was primarily manipulated by varying the amount of sentence to sentence repetition of particular important content words through analysing propositional overlap. Simulating its results with LSA demonstrates the degree to which coherence is carried, or at least reflected, in the continuity of lexical semantics, and shows that LSA correctly captures these effects. However, for these texts, a simpler literal word overlap measure, absent any explicit propositional or LSA analysis, also predicts comprehension very well.

The second set of texts, those from McNamara et al. (1996), manipulates coherence in much subtler ways; often by substituting words and phrases of related meaning but containing different lexical items to provide the conceptual bridges between one sentence and the next. These texts require the detection of the underlying meaning similarities in the absence of literal word repetition. The success of this simulation, and its superiority to direct word overlap predictions, is the principal demonstration of the effectiveness of the LSA coherence measure. This is because LSA relates the semantics of the terms indirectly though the background corpus, showing how LSA was superior to current linguistic methods, which rely on the repetition of keywords to form propositions.

Although LSA performed well in Foltz’s experiments, it has some theoretical limitations in the context of analysing topic flow. This is primarily due to the interpretability of its base vectors, which contain both positive and negative values. In LSA, these base vectors can be interpreted as ‘topics’ whose values describe the extent to which a topic does or does not contain a term. As a document vector is represented in a linear space as a combination of the base vectors it spans, negative values can at times be contradictory in the meaning of a document’s topic mixture. Moreover, this makes it difficult to interpret the topics of a document, as its representation in the linear space is not always a purely additive combination of the topics it spans.

However, this might have a positive effect on performance of LSA in some conditions because it separates the documents into a larger area in k-dimensional
space. This is one reason why LSA has been shown to outperform related topic models in information retrieval and automatic essay assessment.

2.3.2 Non-negative Matrix Factorisation

The interpretability problems with SVD’s basis vectors have led some researchers to propose alternative matrix factorisation methods that maintain the non-negativity of original term-by-document matrix. Non-negative Matrix Factorisation (NMF) is one such method that offers a more intuitive approach for creating a vector-based document model. By only permitting positive entries in its basis vectors, NMF can be used to represent a document as a non-subtractive combination of its parts to form a whole (Lee & Seung, 1999). Unlike SVD, the NMF basis vectors can be more easily interpreted with each giving rise to a distinct topic, allowing for a document to be modelled as an additive non-negative combination of topics. A non-negative solution also allows for topic overlap, and can thus provide a meaningful representation of a document’s topic mixture.

Like SVD, NMF can be used to transform a term-by-document matrix $X$ to a lower rank approximation. This process involves decomposing $X$ into the product of two $k$ rank non-negative matrices, $W$ and $H$, so that $X$ is approximately equal to $X=WH$. The product $WH$ is called a non-negative matrix factorisation of $X$, which can be approximated by minimising the squared error of the Frobenius norm (Meyer, 2000) of $X-WH$. Finding this solution defines the NMF problem, which is mathematically expressed in Equation (2.8).

$$F(W, H) = \|X - WH\|_F^2$$ (2.8)

In the case of document analysis, $k$ can be considered to be the number of latent topics in a document corpus, making the choice of $k$ entirely corpus dependent. An appropriate choice of $k$ remains an open question and as such is often kept constant for individual experiments. The only requirement is that $k$ be chosen such that $k<<\min(m, n)$. Given that $k$ represents the number of latent topics in a document, $W$
becomes a term-by-topic matrix, indicating the weighting of each term in a topic, and $H$ becomes a topic-by-document matrix, indicating the weighting of each topic in a document. This decomposition is mathematically expressed in Equation (2.9).

$$
W \left\langle t_i \right\rangle \\
\downarrow \\
H

X = \begin{bmatrix} w_1 \\ \vdots \\ w_l \end{bmatrix} \begin{bmatrix} h_1 \\ \vdots \\ h_l \end{bmatrix} \leftarrow \left( d_i \right) ^T
$$

Several algorithms have been proposed for solving the NMF problem, including multiplicative update (Lee & Seung, 2001) and projected gradient (Lin, 2007) methods. An NMF algorithm using Euclidean distance minimisation is evaluated in Section 6.2.

In summary, when using vector-based semantic models to measure the coherence of a document, there are a number of technical parameters that need to be considered in order to optimise the analysis results. These include term weighting schemes, distance measures, and dimensionality reduction techniques. These parameters have been shown to influence the semantics of the resulting model significantly.

### 2.3.3 Probabilistic Topic Models

Probabilistic Latent Semantic Analysis (PLSA) is an unsupervised learning method that progressed from LSA and NMF to provide a more robust and theoretically grounded topic model. Like LSA and NMF, PLSA considers text semantics in the form of term co-occurrences; however, unlike LSA and NMF, which use matrix factorisation techniques from linear algebra, PLSA discovers topics by modelling the probability of each co-occurrence as a mixture of conditionally independent multinomial distributions. In the literature, PLSA has been shown to
outperform LSA in information retrieval on a number of document corpora (Hofmann, 1999).

However, despite its claimed superiority over earlier methods, PLSA has been criticised as being incomplete due to it not providing consistent generative semantics (Blei et al., 2003). Using PLSA, a document can be represented as a vector of its topic mixture, but it does not provide a generative probabilistic model for these mixtures. Thus, the number of parameters in the model grows linearly with the size of the corpus, which can lead to severe over-fitting of the data, particularly with smaller datasets. To overcome these theoretical shortcomings, PLSA has been developed into a more robust technique called Latent Dirichlet Allocation (LDA), a probabilistic model that possesses consistent generative semantics. LDA had been shown to result in more reasonable mixtures of topics in a document compared to that of PLSA (Blei et al., 2003).

It has been shown that learning PLSA is equivalent to LDA under a uniform Dirichlet prior distribution (Girolami & Kab, 2003). The LDA and PLSA models can be clearly defined and compared using plate notation, illustrated in Figure 2.4; where the hyper-parameters $\alpha$ and $\beta$ are the uniform Dirichlet prior on the per-document topic distributions and the uniform Dirichlet prior on the per-topic word distribution, $\theta_i$ is the topic distribution for a document $i$, $z_{ij}$ is the topic for a word $w_j$ drawn from the topic distribution for a document $i$. The vector $\theta_i$ of a document $i$ in the LDA model contains a document’s topic mixture. Standard measures can be applied to calculate the distance between the documents’ vectors in the respective topic models.
This problem of learning the various topic distributions in LDA has been solved using algorithms based on variational Bayesian inference (Blei et al., 2003), Gibbs sampling (Griffiths & Steyvers, 2004) and expectation propagation (Minka & La, 2002).

**Theoretical Comparison of Topic Models**

From a theoretical perspective, the probabilistic methods of PLSA and LDA differ significantly from matrix factorisation methods of LSA and NMF. The LSA and NMF build a topic model by approximating a factorisation of a term-by-document. In contrast, PLSA and LDA define generative models of the language to learn the topic model from the dataset. While appropriateness of each method can be debated, the assumptions behind the probabilistic topic models are nevertheless more closely aligned to that of natural language. LDA has some theoretical advantages compared to the LSA and NMF factorisation methods, such as dimensionality selection being unlikely to model over-fitting, and the consistency of the generative model.

In LSA and NMF, term weighting schemes can be used to ameliorate the tendency of over-fitting due to highly frequent terms. Term weighting schemes improve the quality of raw data by eliminating the redundant topical information created by frequent terms. In contrast, PLSA and LDA work naturally on raw frequency counts, making the methods predisposed to give higher weights to the most frequent terms in the corpus. LDA uses hyper-parameters to mitigate this effect of frequent words showing up in all the topics as long as the Dirichlet prior over
topics indicates that some topics occur much more frequently than others do. However, contrary to the findings of information retrieval, LDA has been shown to achieve slightly worse results compared to LSA and PLSA in AES experiments (Kakkonen, Myller, & Sutinen, 2006).

### 2.3.4 Understanding the Topic Semantic Space

The topic modelling methods presented thus far do not take into consideration many important components of a text, such as word order, syntax, morphology, or other features typically associated with text semantics, such as linking words and anaphora. Nevertheless, the topic models built using these methods from the background knowledge of many source documents has long been shown to be reliable in the domain of information retrieval and highly similar to that of humans. Similarly, AES systems have also been shown to have agreement comparable to that of human assessors with the use of pre-scored essays (Wang & Brown, 2007) and course materials (Kakkonen & Sutinen, 2004) as background knowledge.

While these topic modelling methods have been successfully used for information retrieval and AES on a large scale, it could arguably be quite problematic in the case of providing feedback on a single essay. The background corpus used to create a topic model is essentially a set of baseline texts, which are used to define the topic semantics against which a document will be compared. Thus, the distances between the term vectors in a semantic space of a topic model is entirely dependent on the corpus upon which it is built. These relate to the tendency of the words to co-occur in the background corpus. Although manageable, this dependency of the topic model on the background corpus does highlight the importance of its selection, as an inappropriate choice could quite easily lead to errors or biases in the topic model. Such an analysis can bias the meaning of the data, which is arguably undesirable for use in a subjective assessment.

To begin with, any term that does not occur in the original background corpus will not contribute to the analysis, regardless of its importance in a document.
Moreover, the differences in term distributions can be significantly different in a background corpus, compared to the document being analysed. For example, a word that occurs once in a document, but with a high frequency in the original background corpus, is going to contribute very little informational value regardless of the term weighting scheme used. Thus, the term will be ineffective in determining the centroid of the document vector compared to other less important terms in the document.

Similarly, building a topic model on highly technical texts would result in a much more elaborated representation of the semantic space. While this model would accurately capture the semantics of a highly technical document, one would not expect the semantics of a more simplistic document to be as accurately represented in the same model. The background corpus also needs to account for the different ways a term can be expressed in order to identify synonyms and differentiate between the lexical ambiguities of polysemous words. However, if a text contains some implicit knowledge, such a simile or metaphor, this could distort the document representation, and two similar documents could be further apart in the topic model than they should be.

The distances between documents are always going to depend on the knowledge used to create the semantic space. If the knowledge in the background corpus changes then the magnitude of the distances between documents will also change. Thus, a single distance value can be meaningless alone without knowing the proper context of the background knowledge used to create the semantic space. For these reasons, it is important that the background corpus is comprehensive, yet highly domain specific, so that the vocabulary of terms is adequately covered and their distribution is correctly reflected in the context of the domain of the document to be analysed.

**Single Document Semantic Space**

Another approach, suitable for formative assessment, would be to build a topic model with only the sentences or paragraphs of the actual document that is going to
be analysed. This topic model would maximise the variance of the semantic content considered, while focusing on the semantics of the actual document itself, rather than relating it to something external. The approach would be particularly useful for assessing the structure and semantic flow of an essay, as it would provide insight into how a document has been written, rather than comparing it to how other documents have been written in a background corpus.

The idea of constructing a semantic space from a single document was first proposed by Gong and Liu (2001) for the purposes of automatic text summarisation. In their approach, they used the single document semantic space to rank the sentences of a document by identifying the most semantically variant sentences, (i.e., which contain the main topics of the document), while avoiding semantic repetition. The single document semantic space technique has also been used by other researchers for automatic summarisation (Steinberger, Poesio, Kabadjov, & Jeek, 2007) and discovering labels for clustered results in information retrieval (Osinski, 2006).

However, using only the parts of a single document to build a topic model brings a new set of problems to consider. While it is clear that a semantic space built from the document itself maximises the variance and information contained in it. The distances between the parts of the document would no longer be comparable between different documents, nor could they be mapped to a predefined meaning. However, the distances between a single document’s parts would still relative to one another and as such could still be ranked and interpreted accordingly.

To a reader, seeing a word that occurs in one document and seeing a synonym of that word in another document may provide an indication that the two documents are related. Similarly a reader is also able to differentiate between the lexical ambiguities of polysemous words. Mathematically though, such differences can make measuring the amount of topic overlap (i.e., distance) between documents somewhat problematic. While the problems of synonymy and polysemy are a major concern when comparing documents of different authors, such as in information retrieval. Generally this should be less of a problem within a single document as one
would expect the terms used (at least in the case of a well written document) to be fairly consistent. Moreover, given that the topic modelling approaches are reliant on analysing semantics of term co-occurrence rather than understanding their meaning, the actual meaning of the terms can be overlooked in this case.

In some regards, the approach of using a single document semantic space is not as accurate and reliable, as it loses many of the benefits gained from background knowledge. In this sense there is less room for error when using background knowledge. However, the significance of these factors all depends on the application and how the distances are to be interpreted. The single document semantic space is looking at the text in a different way, showing what the writer has done, rather than comparing it to what someone else has done or what they should have done. In addition, to a reader without knowledge of the background corpus, it can be unclear what exactly what the semantic distances are relative to. However, in a single document semantic space, the distances are relative to each other, and thus, a reader only requires knowledge of the document itself to be able to interpret the semantics of the distances.

The two approaches for building semantic spaces are different and the document distances derived from each approach will have different meanings. However, in order to select the appropriate approach, such meaning needs to be defined. The texts themselves are meaningful; they contain the statistical distribution of the words in them that could not be right or wrong. Research in the domain of information retrieval has at least shown the rank order of documents to be meaningful in a semantic space created from background knowledge. In a similar respect, Villalon and Calvo (2009) showed that in the case of document analysis, the rank order of document distances was comparable between semantic spaces created from different background corpora as well as the actual document itself. In this regard, it can be argued that the rank order of the distances is more meaningful when comparing the distances between different semantic spaces. However, in the case of comparing the distances within a single document semantic space, the magnitude of the distances is still the most meaningful feature, simply because it contains more information.
Chapter 3

Visualising the Semantics of Texts

Semantic content is high-dimensional and thus very difficult for a human to interpret alone. The objective is to create a representation of the text with low-dimensionality, so that it can be subjectively interpreted by the end-user. Much of the research on visualising document semantics has been on visualisation in information retrieval based on query and document similarity distances, rather than the semantics of an individual document. These use projection methods to organise documents based on the contents. They are described as feature vectors. Typically, they describe a visualisation in which a document is presented as points on a two-dimensional plane, and the geometric relations of the visualisation points of the documents represent their similarity relations. Such representations are called ‘maps’.

There are a diverse range of algorithms and visual techniques in the information visualisation literature. These typically include a large number of algorithms and parameters, which are largely determined based on the application domain, theoretical grounds or empirical evidence. This chapter outlines relevant algorithms and techniques, and details the decisions and adaptations necessary to apply these techniques for visualising the semantics of texts. Specifically, the chapter examines three types of visualisation techniques; namely, propositional, scatterplot and proximity visualisations.
3.1 Proposition-based Maps

Node-link maps of concepts and propositions are a diagrammatic tool commonly used for representing information as a visual map of semantically related ideas. Such maps have long been used as an educational tool for analysing knowledge through the identification of missing links, misconceptions and false relationships (J.D. Novak, 1998). In the literature, there exists a variety of ‘proposition-based’ maps, which in the general sense, can be regarded as any type of node-link diagram that is distinguished by the use of labelled nodes denoting concepts and links denoting relationships (propositions) between concepts. Such proposition-based node-link diagrams include tools such as concept maps (J.D. Novak & Gowin, 1984), knowledge maps (O’Donnell, Dansereau, & Hall, 2002), cognitive maps, mind maps, semantic networks, and ontologies.

The study of proposition-based mapping as an educational tool was first introduced in Novak and Gowin’s (1984) definition of concept maps. In their definition, a concept map is a hierarchal graph that presents information in a descending order of importance. It consists of concepts and relationships that together form propositions showing how the concepts are linked together. In essence, a proposition is a semantic unit that represents a generalisation of a unit of knowledge.

A concept map describing the keys features of concept maps is shown in Figure 3.1. Each concept generally contains a short noun expression and is represented by a labelled box. Each relationship generally contains a short verb expression and is represented by a labelled line, if the direction of the relationship is down the hierarchy, or by a labelled arrow, if the direction of the relationship is across or up the hierarchy. This defines both the type and direction of the relationship, giving meaning to the resulting proposition. A comparison of some of the many different varieties of proposition-based maps, adapted from the original Novakian model, is given by Åhlberg (2004).
Some educators recommend that it is good practice for students to develop proposition-based maps as a precursor to writing an essay (Straub, 2006). This process helps students to establish the connections between important concepts upon which to structure their argument. Alternatively, by adopting the converse approach, proposition-based maps can also be constructed from the actual essay itself, providing a simple guide to the important concepts and connections in the essay, thus representing the essay in a form that is much easier to analyse and compare against. This approach has been successfully used on a large-scale to carry out an assessment of student essays (Lomask, Baron, Greig, & Harrison, 1992). In this study, proposition-based mapping was shown to be an effective tool for representing and evaluating a learner’s knowledge. However, manually constructing proposition-based maps from text documents can be a tedious, time consuming and challenging process, which can ultimately result in errors or omissions in the final concept map.

There are many software tools designed to assist users in constructing proposition-based maps. Mapping tools, such as CmapTools (Cañas et al., 2004), help to reduce many of the obstacles in constructing complex maps and allowing users to share concept maps, and participate in an integrated threaded-messaging forum for
the discussion of a particular concept map. Other tools, such as Verified Concept Mapper (Cimolino & Kay, 2002), aim to support the construction on maps by providing feedback to aid reflection and as a basis to model a learner’s understanding.

The appeal of proposition-based mapping is in its ability to represent information concisely and explicitly. Visualising information as a map has been suggested ‘to take advantage of the remarkable capabilities of the human visual perception system and the benefits of visual information representation. These benefits include (a) ease of recognition, (b) the possibility to quickly scan a picture and find differences or keywords, (c) compactness of representation, and (d) the observation that it seems to be easier to keep an overview’ (Kommers, 1997). There are also many structural properties that have been shown to greatly influence the cognitive processing of proposition-based maps, such as spatial configuration, map format and link structure (Wiegmann, Dansereau, McCagg, Rewey, & Pitre, 1992). It is suggested that a good proposition-based map should contain approximately fifteen to 20 concepts and ten to fifteen relationships in order to adequately represent an individual’s knowledge of a particular topic (J.D. Novak, 1998). This also provides an adequate size for representing the knowledge in a short essay.

In order to infer a proposition-based map from a text, the degree of semantic relatedness between pairs of concepts needs to be measured. Danowski (1993) presents a completely automated approach for mapping text in which a link is added between each pair of words within a specified window size. The window size defines the number of adjacent words that any linked pair needs to fall within. While this approach has the advantage of being completely automated, its resulting map is unfiltered and treats every word as a distinct concept. Using this approach, a short text of only a few hundred words would produce a map with an overwhelming amount of concepts, far exceeding Novak’s ideal concept map size of fifteen to 20 concepts. Carley (1993) addressed this issue, by extending the windowing approach and introducing a number of statement formation coding options. The coding options allow the user to determine which concepts to include in the map and how their degree of relatedness is determined. Others have used statistical measures of
the word co-occurrence (Matsuo & Ishizuka, 2004) and centrality measures (Palshikar, 2007). More advanced approaches, such as a Naive Bayes classifier (Domingos & Pazzani, 1997), use a supervised learning method. However, a supervised approach can sometimes be infeasible, as it requires a large domain specific dataset for training.

Automap is a textual analysis tool that uses a windowing approach to extract proposition-based maps from texts. While some of the methods used by Automap are semi-automated, such as stemming and deletion, others, such as generalisation, required a predefined thesaurus that specifically denotes which concepts to generalise for each individual text. While generalisation can be somewhat automated through the use of a thesaurus, such as WordNet (Miller, 1995), Automap does not support such a feature. Automap also provides no way of visualising the shared knowledge of multiple texts and authors.

In summary, the methods presented thus far provide an overview of proposition-based models for visualising the semantics of texts. These visualisations are primarily concerned with the semantics of a text’s content as whole. Although such approaches may be useful for identifying gaps or misconceptions in content, they provide little indication as to the quality of its coherent flow, as they give no indication of sequence.

### 3.2 Scatterplot-based Maps

Topic models provide an effective technique for modelling texts as a mixture of their semantic content for visualisation. This section examines a number of techniques for implementing such applications of topic models. These applications explore the relatedness of documents by representing the principal dimensions and the semantic similarity between objects to create two-dimensional scatterplot visualisations. This section presents a number of such approaches for visualising text documents.
A common mapping technique for analysing the relatedness of documents involves exploring the semantic space on a dimension-by-dimension basis. Landauer et al. (2004) used Latent Semantic Analysis (LSA) to visualise the semantic relatedness of documents, by exploring three principal dimensions of the semantic space. LSA was used to analyse a text corpus containing 16,169 articles papers from the Proceedings of the National Academy of Science (PNAS) volumes 94–99. To create their semantic space, the text corpus was divided up into 317,115 paragraph-like passages that included the full title, abstract, and body text of all articles. Each entry in the term-by-document occurrence matrix was weighted using the measure of Log-Entropy (i.e., the log of the frequency of a term in a paragraph weighted inversely with the entropy of the word across all paragraphs). After performing the LSA and constructing the semantic space, the vector representation of each PNAS article was constructed from the sum of their term vectors.

Each PNAS article vector was then represented in a three-dimensional space. In their case, the authors used the combination of dimensions three, four, and five, which they found to be of greatest interest. However, as only three out of hundreds of dimensions are being considered, the visualisations did exhibit a substantial amount of information loss and were thus quite limited in the semantic features they could reveal. In an attempt to alleviate this effect, the authors use a high-dimensional dynamic viewer tool called GGobi (Swayne, Lang, Buja, & Cook, 2003) to explore and help uncover patterns of interest in the underlying data. GGobi allows for a high level of user control to explore revealing views and identify points of interest. Figure 3.2 depicts the topic clusters of the documents from a visualisation of all articles from eight biology subfield categories in PNAS. The visualisation shows the initial algorithmically chosen view (Left) and an exploratory user selected view (Right). In subsequent visualisation, the authors introduce the measure of cosine similarity to uncover documents related to a specific query. The results show that this mapping approach is capable of keeping most of the documents of the same categories in the same region of the map. However, there is of course some overlap expressed in the map due to the presence of similar concepts across these categories.
Figure 3.2: PNAS articles coloured according to their biology subfield categories. The two-dimensional view on the three-dimensional space was selected algorithmically (Left) and by aided human selection (Right). Figure reproduced from Landauer et al. (2004).

Vande et al. (2008) used LSA to visualise the dynamics of team collaboration. In the study, the knowledge of each participant was extracted from a text corpus consisting of transcripts of verbal dialogue, where each utterance is equivalent to a single document. This was used to build a semantic space and measure the semantic coherence of the shared knowledge of team participants. In their implementation of LSA, they use the Stanford Natural Language Processing Parser to extract terms for the text corpus, where a term is considered to be either a word or noun phrase (i.e., a phrase that begins with a noun or pronoun). These terms and documents were then used to create an occurrence matrix where each term is weighted by its frequency of occurrence in a corresponding document. After generating the semantic space from the resulting matrix, the knowledge of each participant could then be represented in the semantic space by the aggregated vectors of their corresponding documents at any point in time.
To compare this resulting knowledge of team participants, the measure of cosine similarity was used to calculate the semantic coherence between the knowledge vectors of participants. In each calculation, the conceptual coherence of a participant’s knowledge was measured relative to a static centroid calculated from the average of all document vectors. Therefore, the knowledge of a participant at any point in time is measured by the semantic coherence of the participant’s text to the final shared knowledge of the team, thus allowing the dynamics of the teams shared knowledge to be compared over the course of the project.

In order to express these knowledge comparisons in a meaningful way, a scatterplot was used to represent the dynamics of semantic coherence of the teams shared knowledge over the course of the project. This type of representation allows an end-user to track the convergence and divergence of the knowledge of all participants relative to the shared knowledge of the team at any point in the project. In the scatterplot visualisation example shown in Figure 3.3, the conceptual coherence of a team of six participants who contributed 1,000 content bearing utterances consisting of 615 unique noun phrases. Five of the teams’ members converge to a common coherence, the coherence of, indicating the Person-E is contributing less to the shared knowledge of the teams.
Figure 3.3: A scatterplot of the conceptual coherence of team participants. Figure reproduced from Vande et al. (2008).

The context of Vande’s visualisations was for illustrating the dynamics of a team’s conceptual coherence in order for an observer to understand and infer team cohesiveness. The authors did this by analysing text. Instead of the texts of teams of people, the same technique could be used to visualise how the coherence of an individual document’s parts (i.e., sentences, paragraphs) changes over the course of a text.

While this scatterplot method is commonly used for text document visualisation, it falls short when applied for text document visualisation. One of problems with only visualising the selected dimensions of a semantic space is that at most only two or three dimensions can be analysed at any one time. Thus, the information in the remaining dimensions is never considered. Which often leads to significant information loss (Huang, Ward, & Rundensteiner, 2005). Moreover, it is not always clear what the visualised dimensions actually mean.
3.3 Proximity-based Maps

Proximity visualisation algorithms aim to preserve the topology of the relationships of a dataset by converting the complex relationships between high-dimensional objects into simple geometric relationships in a low-dimensional space. This process can be used to form a two-dimensional map in which similar objects are positioned closer together and dissimilar objects further apart. One of the advantages of proximity visualisation over other visualisation techniques is that it is independent of the number of object variables. All that is required is to ascertain the high-dimensional distance between objects.

This type of data arises when the relationships between objects in a dataset can only be more accurately represented by measuring how similar or dissimilar they are, as opposed to just a simple comparison of their commonalities. This approach is ideal for document analysis, as documents in a vector space model typically contain a high dimensional mixture of topics. A useful measure for document dissimilarity is defined in Equation (3.1); where $s_{ij}$ is the similarity between documents $i$ and $j$.

$$\delta_{ij} = \sqrt{1 - s_{ij}} \quad (3.1)$$

Two proximity visualisation methods are presented here: Self-Organising Maps (SOM) (Kohonen, 2001), which map the object relationships to a grid; and Multidimensional Scaling (MDS) (Borg & Groenen, 2005), which map the object relationships to a Euclidean space. Two of the major approaches have which been applied in the literature as an approach for clustering and visualising text document collections (Huang et al., 2005).
Self-organising Maps

Self-organising Maps (SOM) is an approach for mapping object similarities to the cells of a rectangular grid. SOM implements a neighbourhood function, which attempts to preserve the spatial properties of the distances between a dataset’s objects (Kohonen, 2001). The purpose of the SOM algorithm is to position each object meaningfully in a separated grid cell, such that similar objects occupy neighbouring cells, and are distant from cells of dissimilar objects. The effect of transforming the representation to a ‘proximity’ grid ensures that a minimum amount of separation is maintained around an object, while still keeping similar objects together.

One of the obvious benefits of associating objects with the cells of the grid is that there is no possibility of visual overlap. The SOM approach allows for all objects, even those in high density clusters, to be clearly visualised. This is in contrast to other proximity visualisation techniques, such as MDS, which do not take the actual size of the dataset objects into consideration. Using such techniques can lead to a partial or complete overlap when representing the dataset objects, which has previously been attributed as a problem when performing visually-based user tasks (Rodden, Basalaj, Sinclair, & Wood, 1999).

However, the advantage of greater clarity in regards to eliminating visual overlap comes at the cost of increased errors in the resulting SOM visualisation. In addition, the actual meaning of the distances between objects changes, as there is no exact quantitative value that can be attributed to one or more cell intervals. Moreover, in the context of visualising a single document, the SOM approach could be quite problematic given that the different parts of a text (for example, sentences or paragraph) can significantly vary in size. This could result in some parts having disproportionately large cells, which could ultimately distort the final visual representation. Further, using SOM could be in some cases detrimental in regards to providing feedback. For example, paragraphs in a document that overlap are likely to be very similar in content, which may in itself indicate something of importance in regards to feedback.
3.3.1 Multidimensional Scaling

Multidimensional Scaling (MDS) refers to a class of algorithms for representing similarities between objects (Borg & Groenen, 2005). The objective of MDS is to reduce the observed complexity of high-dimensional objects, by spatially representing them as points in a low-dimensional space (typically two or three dimensions). MDS is most suitable for those application areas where every dimension has a scale, and the distances are meaningful, such as the distances in a document corpus in a VSM model. For example, given the distances between all the cities in a country, MDS could be used to plot the relative location of each city on a map. In a similar way, MDS could be also used reduce the observed complexity of a document’s high-dimensional topic mixture to a low two-dimensional map-like representation for visual consumption.

All MDS algorithms typically start with a matrix of object-by-object dissimilarities $\delta_{ij}$ as input. There is a magnitude of MDS algorithms available that are generally categorised as being either Metric or Non-Metric. Metric MDS algorithms compare distances between objects directly and attempt to project this relationship in their dimensional transformation. Conversely, non-metric MDS algorithms are concerned with projecting the rank order of objects rather than the distances by performing a monotonic transformation on the data. Additionally, researchers must decide on the number of dimensions they want the computer to create. The more dimensions, the better the statistical fit, but the more difficult it is to interpret the results.

The final MDS configuration results in a representation of the documents such that the distance between the points in the two-dimensional space resemble as closely as possible the dissimilarities between the documents in the high-dimensional space. In any configuration of points, the similarity between the documents is solely represented by their relative proximity, with actual orientation of the axes being arbitrary. However, in any configuration, the optimal minimisation of the loss function is ultimately restricted by the complexity of the dataset. A typical loss function in this case is Energy (others can also be used). Energy can be expressed as
the sum of the normalised squared errors between the document dissimilarities $\hat{d}_{ij}$ and their approximated distances $d_{ij}$ in the low-dimensional space. The mathematical formula for calculating energy is defined in Equation (3.2).

$$\sigma_E = \sum_{i<j} \left( \frac{d_{ij} - \hat{d}_{ij}}{\hat{d}_{ij}} \right)^2$$  (3.2)

However, this measure is a purely summative combination of the errors in a configuration, making it unsatisfactory for comparing configurations of different size. Alternatively, the energy can be normalised by the number of distance values in a configuration. This adaptation to the formula makes the loss function independent of the size of the configuration, making it is possible to compare configurations of different documents. The normalised energy measure is mathematically defined in Equation (3.3), and will give a value between 0 (perfect fit) and 1 (worst fit).

$$\sigma_{En} = \frac{2}{n(n-1)} \sum_{i<j} \frac{n^2}{\hat{d}_{ij}} \left( \frac{d_{ij} - \hat{d}_{ij}}{\hat{d}_{ij}} \right)^2$$  (3.3)

A document consisting of $n$ paragraphs will have a total of $n(n-1)/2$ distance values for consideration. This means that as the number of paragraph in a document increases, the number of distance values to consider will grow at a quadratic rate. Although unique for each document, in general the fit error for an MDS configuration will be most affected the number of paragraphs in a document along with the number of dimensions used in the semantic space to present the document. However, it is not certain where this point is exactly or how much error is tolerable for the paragraph mapping approach to be remain accurate.

Mathematically, as the number of objects in a high-dimensional space increases, the energy in the low-dimensional MDS model will either increase or remain the same. Thus, for any given document, it is most often impossible to represent it perfectly in two dimensions. However, it is not necessarily the case that an MDS map needs to have zero energy in order is meaningful. Of course, a certain amount of
error is tolerable. Finding this point and the limits of such a representation is an open question.

In the MDS literature, various error tolerance levels have been suggested for different loss functions. In particular, the quality judgements proposed by Kruskal (1964) are often referenced and used as a general guideline (see Table 3.1). The quality judgement of an MDS configuration is subjective and problem dependent, and indeed Kruskal’s guidelines are based on his own experimental experience.

Table 3.1: A guideline of quality judgements for MDS configurations proposed by Kruskal (1964).

<table>
<thead>
<tr>
<th>Quality of configuration</th>
<th>Error of fit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poor</td>
<td>20.0%</td>
</tr>
<tr>
<td>Fair</td>
<td>10.0%</td>
</tr>
<tr>
<td>Good</td>
<td>5.0%</td>
</tr>
<tr>
<td>Excellent</td>
<td>2.5%</td>
</tr>
<tr>
<td>Perfect</td>
<td>0.0%</td>
</tr>
</tbody>
</table>

3.3.2 Classic Multidimensional Scaling

Classical multidimensional scaling, also known as Principal Coordinate Analysis (PCO), attempts to solve the MDS problem by taking the input matrix giving dissimilarity matrix $D$ and outputs a coordinate matrix whose configuration attempts to minimise the loss function (Borg & Groenen, 2005). Classical MDS finds a unique solution by performing eigen-decomposition on a dissimilarity matrix, as outlined Gower (1966). Gower’s centred matrix $G$ is calculated in Equation (3.4); where the elements of $A$ represent the centred elements of the distance matrix $D$, as shown in Equation (3.5), and where $1$ is a column of 1’s of length $n$ and $I$ is the identity matrix. The matrix $G$ is then decomposed into its component eigenvalues and eigenvectors. The eigenvectors, normalised by dividing by the square root of their corresponding
eigenvalue, are output as the principal coordinate axes for the configuration of points.

\[ G = \left( I - \frac{1}{n} \hat{1} \hat{1}^T \right) A \left( I - \frac{1}{n} \hat{1} \hat{1}^T \right) \]  

(3.4)

\[ a_{ij} = \left( 1 - \frac{1}{2} d_{ij}^2 \right) \]  

(3.5)

PCO has been experimentally shown to be one of the fastest, but least accurate, of the MDS algorithms. However, the approach is quite reliable in the sense that the algorithm will always produce the same configuration of points for a given distance matrix. Consequently, it has been suggested that PCO is better suited to a seeding method to provide a starting configuration for other more advantaged multidimensional scaling methods (Basalaj, 2001).

3.3.3 Metric Multidimensional Scaling

Metric MDS offers an improvement over classical MDS by optimising the projection procedure to a loss function, such as that of Energy in Equation (3.2). Metric MDS algorithms typically undertake an interactive approach to minimising the loss function until convergence occurs.

A prototypical algorithm is described as follows (Borg & Groenen, 2005):

1. Initialise a configuration \( X \) of points.

2. Find \( d_{ij} \) from the configuration of points.

3. Find a new configuration \( X_{n+1} \), which minimises the loss function (metric step).

4. Evaluate the fit of the distances \( d_{ij} \) and the disparities \( \hat{d}_{ij} \).
Many procedures can be used to minimise the loss function. For example, Kruskal recommended an iterative steepest descent approach. However, De Leeuw (1977) introduced a significantly better procedure for minimising the loss function, in terms of speed and likelihood of convergence. De Leeuw’s iterative majorisation method minimises a simple majorising function. This iterative majorisation process is also commonly referred to as the SMACOF algorithm (Scaling by Majorising a Complicated Function).

An evaluation of the performance of some of the most commonly used metric MDS algorithms is presented in Basalaj (2001); namely, Newton-Raphson, Tabu Search, Genetic Algorithm, Iterative Majorisation, and Simulated Annealing. These MDS algorithms undertake an iterative, least-squares approach to MDS. These algorithms attempt to relate the cosine dissimilarity between high-dimensional objects to their Euclidean distance in a low-dimensional space by minimising a loss function. The distances in the Euclidean space allows humans gauge the magnitude of the document similarity.

3.3.4 Non-metric Multidimensional Scaling

In some other applications, the measured distances may not necessarily have a clear meaning. Thus, the magnitude of the distances may be deemed unreliable, and only the relative distinction in their order holds meaning. In this case, the distances should be replaced by their rank order. Such a procedure will cancel the effect of distance magnitude, which could otherwise greatly distort the visual representation.

Similar to metric MDS, the objective of non-metric MDS is to create a spatial representation of the objects with low-dimensionality, but based on the rank order of dissimilarities rather than quantities themselves. The non-metric MDS algorithm undertakes the same procedure as metric MDS, except for the addition of a non-metric step.
The prototypical algorithm for non-metric MDS, developed by Shepard and Kruskal (Borg & Groenen, 2005), is described as follows:

1. Initialise a configuration $X$ of points.
2. Find $d_{ij}$ from the configuration of points.
3. Determine $\hat{d}_{ij}$ using the PAV algorithm (non-metric step).
4. Find a new configuration $X_{n+1}$, which minimises the loss function (metric step).
5. Evaluate the fit of the distances $d_{ij}$ and the disparities $\hat{d}_{ij}$.

The non-metric step, Step 3, determines the disparities $\hat{d}_{ij}$ from the distances $d_{ij}$ by constructing a monotone regression relationship between disparities $\hat{d}_{ij}$ and dissimilarities $\delta_{ij}$, under the weak monotonic requirement, which provides for the fitting of unequal data by equal or unequal pseudo-distances, as described in Equation (3.6).

$$\delta_{ij} < \delta_{kl} \Leftrightarrow \hat{d}_{ij} \leq \hat{d}_{kl} \quad (3.6)$$

In order to obtain the disparities $\hat{d}_{ij}$, an approximation method called the Pool Adjacent Violators (PAV) algorithm is commonly used. In the PAV algorithm, starting with the lowest ranked dissimilarity $\delta_{ij}$, each dissimilarity $\delta_{ij}$ and its adjacent distance $d_{ij}$ are compared to determine if they are monotonically related. If the required monotonic property is true, then the disparity $\hat{d}_{ij}$ is set to distance $d_{ij}$. Whenever a block of consecutive values distances $d_{ij}$ are encountered that violate the required monotonic property the distances are averaged together with the most
recent non-violator distance $d_{ij}$ to obtain an estimated disparity $\hat{d}_{ij}$. Following the non-metric step, the algorithm proceeds using the same techniques as metric MDS to obtain a spatial configuration of points such that the distances $d_{ij}$ are more closely resembles the approximated new disparities $\hat{d}_{ij}$.

The rank order of distances is an important feature to consider when interpreting the semantic structure and flow of an essay, as their magnitude and meaning can differ from one semantic space to the next. That is why comparing the ranks and in the case of MDS a rank preserving instead of a distance preserving version can sometimes be desirable. However, this brings a new interpretation to the final MDS projection. It is important to understand how this changes the representation of the data. What exactly do the visual representations mean in a distance preserving or rank preserving MDS? Clearly, they both provide some insight into the semantics of an essay, but whether the distance magnitude or rank order is more important is still an unanswered question. Based on a study by Villalon and Calvo (2009), however, it can be argued that in the case of document analysis the rank order of the distances is more important when comparing the distances between semantic spaces, but the magnitude is more important when comparing the distances within a single space.
Chapter 4

Glosser - Automatic Semantic Feedback Tool

The research presented in this thesis contributed to the development of an automated feedback tool called Glosser (Villalon et al., 2008). Glosser has thus far been used by hundreds of university students over a two-year period, from 2008 to 2009, and continues to receive financial support through various grants until at least 2010. Glosser is intended to facilitate the review of argumentative essays by supplying useful information about the essay, which is intended to make the evaluation of the essay an easier task. It does this by providing a number of tools for analysis and exploring the semantics of an essay.

Glosser was designed as a framework to provide a basic conceptual structure for supporting reflection in academic writing. Glosser scaffolds reflection through the use of text mining techniques to provide automated feedback in combination with a set of trigger questions to help prompt reflection. The framework provides an extensible plugin architecture, which allows for new feedback tools to be easily developed and put into production.

The user interface of the Topics tool in Glosser is displayed in Figure 4.1. The questions at the top of each page are provided to help the reader focus their evaluation of the essay. They are intended to make sure that students cover all aspects of a good argument related to the features presented. Below the questions is the supportive content called ‘gloss’, to help the reader answer those questions. The ‘gloss’ is the important features that Glosser has highlighted in the essay for reflection.
Figure 4.1: The user interface of the Topics tool in Glosser. The trigger questions are displayed at the top of the page and ‘gloss’ below.

The tools do not give a definite answer, rather they tell a user what the writer has done, and not what they should do. Glosser simply highlights features for reflection that are related to common problems in academic writing. It is up to the user to decide whether the highlighted feature has been appropriately used. The reader should use the programme accordingly, and use the information provided by the programme to help review the essay. Overall, however, the user should focus on making the essay more structured, comprehensive and readable.

Glosser carries a number of assumptions in presenting the feedback questions and ‘gloss’. Firstly, it assumes that essays should be an argument, asserting a particular point of view, and should support this argument with well thought out
reasoning and evidence. It should not be a re-stating of official, academic knowledge, or a simple, unsupported opinion. Essays should be comprehensive, while remaining relevant, when discussing a topic. These assumptions are widely held, both in academic research and writing departments The second assumption is that Glosser cannot decide whether the evidence and argumentation is reasonable or not. That will always be something that must be decided by another reader. However, the feedback provided by Glosser can help this evaluation by providing the reader with information about the sort of statements the essay is making, the topics and keywords it includes, and the way these are organised.

4.1 Design and Architecture

Glosser was designed to support writing activities by real students, as opposed to a prototype or proof of concept application. Real applications present many challenges in terms of scalability and concurrency. The architecture and technologies used in Glosser have been chosen with the specific purpose of addressing such problems. This section explains both the design decisions behind these choices and demonstrates how to extend the framework to develop new feedback tools.

Glosser was built using the Java programming language, largely due to the availability of open source libraries and frameworks to support its development. Glosser was developed as a web application for several reasons. Firstly, a web application has the advantage of being easy to update and maintain an installation of Glosser without having to distribute and install the software on client computers. Secondly, it has cross-platform compatibility, assuming a client’s web browser complies with the relevant W3C web standards. Finally and most importantly, the text mining operations that Glosser performs on an essay are extremely hardware intensive, and are thus are required to be run on a high performance server.

Glosser was developed using the Spring framework, which implements the Model-View-Controller (MVC) architectural pattern. The MVC pattern separates the
application into three tiers by isolating business logic from user interface considerations, resulting in an application where it is easier to modify either the visual appearance of the application or the underlying business rules without affecting the other. In MVC, the model represents the information (the data) of the application; the view corresponds to elements of the user interface, which in this case are JSP, and the controller manages the communication of data and the business rules used to manipulate the data to and from the model. This framework offers several advantages in terms of scalability as well as maintainability of the code.

The Text Mining Library (TML) package was used in Glosser to benefit from its comprehensive set of data mining algorithms. TML provides Glosser with a text mining infrastructure of help scaffold every stages of the text mining process. TML is an open source package, which implements text mining specific algorithms from the Stanford NLP and Weka machine learning library (Witten & Frank, 2005). At the core of TML is the Apache Lucene search engine. The Apache Lucene search engine is used in enterprise level applications. It provides functionalities for the pre-processing of documents, tokenising, stemming and removing stopwords, as well as part-of-speech tagging to improve the selection of terms for analysis further. TML creates three corpora for each new document, at the sentence, paragraph and document level. TML stores a model of a document in a Lucene index, as well as storing the results of the text mining operations.

The Prefuse visualisation toolkit was used in Glosser to benefit from its comprehensive set of algorithms for graph-based visualisations. Prefuse is a Java-based visualisation toolkit that was developed by Heer (2005) and written using the Java2D graphics library. Prefuse provides an extensive set of features to support visualisation, animation, and interaction, greatly simplifying the process of developing complex visualisation. It supports a host of data structures, layouts, renders and filters as well as integrated searching of data and database connectivity. The design of Prefuse allows developers to implement the functionality they require easily, and customise and extend it to meet their own needs. Prefuse was used by Glosser to build document visualisations, which could then be presented as PNG images to the end-user.
An important feature of Glosser is that it does not manage users, documents or permissions. These aspects present their own set of complex problems, which are better solved by other providers. It is not in our interests. Instead, we focus our attention on integrating existing services to make them accessible to Glosser.

Glosser uses the WASM authentication service to manage user accounts. WASM is a web single sign on system developed at the University of Sydney. It allows a number of different web-based applications to use a single login for their University of Sydney student/staff account. It thus allows for the seamless integrations of a student’s existing account with Glosser. Similarly, the documents and permissions are managed using an external service, but downloaded and used by Glosser as required. The design is shown in the figure, followed by a brief outline of the function of each class.

A class diagram outlining the structure of Glosser’s extensible architecture is shown in Figure 4.2. The main entities are the Configuration, Site, Harvester Repository, Indexer and Tool. Below is a description of the purpose and functionality of each entity, followed by a section describing the implementation of the Tool entity in Glosser.

![Class Diagram](image)

Figure 4.2: A class diagram outlining the structure of Glosser’s extensible architecture.
**Configuration**

The Configuration entity contains the specific settings for an installation of Glosser, such as the database and proxy. Additionally, it also includes the configurations for each individual Site. These settings can be modified through an admin interface.

**Site**

A Site entity contains the specific configuration for each unit of study course. This configuration includes the list of included tools, the language locale used, any course specific messages (for example, welcome message and alternative tool questions), and type of writing tool from which the documents need to be harvested (downloaded).

**Harvester**

As the word processing tool used to write a document is managed by an external service, separate to Glosser, there was a need for an a standardised interface to encapsulate the functionality for accessing the document information. This interface is called a Harvester. Each site requires a specific Harvester to download the document details and content, store them in a database, and manage the permissions for each document. In developing Glosser, we have experimented with a number of writing tools for which Harvesters have been developed, including Trac wiki, Google Docs, Moodle wiki and Sakai wiki.

**Repository**

There was a need for an entity similar to that of a database to store documents of various formats in a generic way. This functionality is subsumed in Glosser in a so-called Repository entity, which represents a Lucene index in TML. The Lucene index
is used to clean and store a document along with its sentence and paragraph corpora from which the text mining operations can be performed.

**Indexer**

Glosser allows users to download and index a new revision of a document on demand, at a time their convenience. However, Lucene locks the index when inserting a new document. This can cause concurrency problems if multiple processes are attempting to insert a document at the same time. To deal with the concurrency problems, the execution of document insertion has been decoupled from rest of the application. Instead, a single Indexer thread is run for each site to manage the queue of documents for harvesting and inserting into the Lucene index. A separate thread that waits until there are new documents queued in the database before proceeding to insert them into the Lucene index one at a time.

![Diagram of the index process in Glosser](image)

**Figure 4.3:** A data flow diagram describing the process of harvesting and indexing a document in Glosser.
4.2 Feedback Tools

A ‘tool’ in Glosser defines a page with a set of gloss and questions. Each tool provides a different way for a user to analyse a specific feature of a document. Glosser provides a simple and extensible framework for providing a standard way to create additional tools. The framework allows new tools to be seamlessly added to Glosser, without any modification to the original source code. The procedure for installing a new tool simply requires its Java classes and associated dependencies to be added to the Glosser class-path.

Creating a tool requires the Annotation of a single Controller class, which implements the business logic of the tool and returns a view of the tool ‘gloss’. Annotations are feature of the Java programming language, which provides a convenient way of storing metadata in a class. In this case,Annotations are used to define how Glosser should interact with a tool class. The metadata stored in a tool class is read by Glosser at runtime and made available to use in a Glosser installation.

Figure 4.4 provides an example of how to make a simple spell-checker Tool, which was implemented using the GNU Aspell library. The resulting ‘gloss’ returned by the tool class is wrapped in the Glosser template using the Apache Tiles framework. This maintains the look and feel of Glosser across all tools.
@Controller
public class SpellCheckController extends ToolController
{
    @RequestMapping("/spellcheck.htm")
    public ModelAndView handleRequest() throws Exception
    {
        // initialise spell checker
        SpellChecker spellchecker = new SpellChecker();
        spellchecker.initialize(dictionary);

        // spell check document html
        String errors = spellchecker.spellCheckXML(doc.htmlText());

        // return model and view
        Map<String, Object> model = new HashMap<String, Object>();
        model.put("gloss", errors);

        return new ModelAndView("spellcheck", model);
    }
}

Figure 4.4: A Java class exemplifying how to create simple spell-checking tool in Glosser.

A number of tools within Glosser were developed to target specific problems in writing. These include tools for analysing surface features, such as the word count and spelling and grammar errors, as well as more complex tools for analysing the semantic flow of the text. This section describes some of the tools implemented in Glosser for analysing the semantics of writing.

Writing Style Slider

This tool indicates whether two paragraphs are written in dissimilar writing styles. This is measured purely on the word and sentence length of the subject matter as defined using the Readability Ease in Equation (2.1). Thus, it ignores other ways of measuring writing style, such as the actual meaning of the subject matter. If two paragraphs are in bold, it means that there has been a large jump in the style of writing used. If this makes the essay difficult to read or understand, then it is suggest to the reader that it should be changed. The slider bar is used to adjust the threshold for the writing style warning. An example of the output from this tool is displayed is
Figure 4.5. The icon suggests there is a possible shift in the writing style between consecutive paragraphs.

We see evidence in every day life that English is being spoken by more non-native English speakers and that more people are taking-up English as a second language. However according to the U.S. Bureau of the Census, states that due to the rise of immigration to the US between 1980 and 1990 Spanish speakers increased by 50 percent, and out of “2.4 Chinese speakers in the US more than four out of five speak Chinese at home” (Walrafen, 2000).

Moreover, if statistics could prove that there was a definite increase in the English population, it would not prove that it is the lingua franca. Perhaps as far as international business is concerned; English might be considered the universal language. However, if you were to take a round the world holiday and visit many diverse countries and communities, then English would be a far cry from a universal communication code.

When looking at the uprisings of English speakers, there are possibly many advantages, especially for those who speak it as a first language; but I wonder do the advantages outweigh the disadvantages; via the gatekeeping mechanism (Greedid, 2000). Greedid suggests that proper knowledge of English will bar some people from certain jobs. If this is a case, in most factories, it means that people would be denied jobs, even when they apply for mental process work, and positions that do not require a large vocabulary in any language.

So, in conclusion, English does appear to be on the uprisings but not everywhere. There is also advantages and disadvantages. I would like to personally quote Douglas Adams on the dangers of a universal language. Concerning his fictional Babel Fish; a device that translates any language into the vernacular of the user, by just sticking it in your ear: “The Babel Fish has caused more... and bloodier wars” (Adams, 1977). Exactly what Adams means by this I am not completely sure, but perhaps he means that if every government could understand each other perfectly, there would be total chaos.

Figure 4.5: An example of the ‘gloss’ output produced using the Readability Slider tool. A warning icon is displayed when the difference in writing style between two paragraphs is below the threshold value of the slider.

Coherence Slider

This tool indicates if two paragraphs are incoherent. This is measured purely on the similarity of the subject matter, and it ignores other ways of producing flow, such as linking words (for example, ‘however’, ‘first’ and ‘secondly’). If two paragraphs are in bold, it means that there has been a large jump in the subject matter. If this makes the essay difficult to read or understand, then it should be changed. The slider bar is used to adjust the threshold for the coherence warning type. An example of the output from this tool is displayed is Figure 4.6. The icon suggests there is a possible break in topic between consecutive paragraphs.
In the 21st century, English has been seen and spoken many places in the world. Yet, it does not mean English is the global language. Many parts of the world are still speaking their own languages and English is their second or third language. One example is Hong Kong, a former British colony. Students learn English since they are in kindergarten, they never stop learning/using English until they have a job. At the meantime, local Hong Kong students speak Cantonese most of their life in a day, the only English speaking time is school’s English class. It proves that people learn English but they do not use it very often. In any English speaking nations like the United States, there is increasing of immigrants every year. Many of them are from South America and Asia. In which most of them may not speak English very well but they do not find any problems and not even bothered of learning it. This scenario shows that the globalization of English will never happen because not everyone uses English in his/her life.

As English is increasingly used in the world, it results a positive development for countries and its people. It helps Eastern countries to have business & trades with the western, it can prosper both countries. And since English is a second/third language in the world, people can communicate with each other across nations, races and ethnic. English can help countries to get closer by breaking the language barrier.

In conclusion, there is no doubt that the use of English is increasing and it is a positive development. Though English hasn't reach the stage of being a global language/ because some other languages like Chinese, Spanish and French speakers are also increasing. One of each may become the global language in the future.

Figure 4.6: An example of the ‘gloss’ output produced using the Coherence Slider tool. A warning icon is displayed when the topic overlap between two paragraphs is below the threshold value of the slider.

Topic Flow Map

Topic flow refers to the average amount of semantic overlap between paragraphs. We believe that there are cases in which it would useful to be aware of the ‘semantic’ development of a document. This tool attempts to position paragraphs in a two-dimensional map based on the similarity of the topic content discussed in the essay. It is also an opportunity to consider the cohesiveness of the essay.

The map displays the paragraphs from each topic as they are used in relationship to each other. If two paragraphs contain similar topic mixtures, they are positioned more closely in the representation. For example, the topics covered in a text may not be clear or explicit. Getting a better sense of the semantic structure and flow of these topics could allow for a more coherent text. This map forms one of the main contributions of this thesis and is discussed in more detail in Chapter 5. An example topic flow map is provided in Figure 4.7.
Figure 4.7: A topic flow map consisting of five paragraphs. The similarity of the paragraphs’ topic content is represented by their relative proximity in the map.

**Topic Coverage Map**

Topic coverage refers to the semantics topics mentioned in the text. We believe that there are cases in which it would useful to be aware of the ‘semantics’ being introduced and developed in the document. The tool attempts to list the topics that have been discussed in the essay, and the sentences that contribute to those topics. It is an opportunity to consider the comprehensiveness of the essay, whether it has included everything it should to cover the topic well, and excluded anything that is not relevant.

The map displays the keywords from each topic as they are used in relationship to each other. If two keywords are used in the same sentence, they are linked in the
network representation. For example, the topics to be covered may not yet be clear or explicit. Getting a better sense of what these topic areas are could allow for a better distribution of work. A good network should link a variety of keywords, not just the central word to the others. That is, a good essay should be discussing the concepts from as many angles as possible, comparing them with other concepts, and not simply stating them on their own.

The topic coverage map is similar to the standard proposition-based knowledge map, where nodes represent concepts and the links represent the relationships between these concepts. However, it also includes an additional layer, which relates the semantics of the essay’s topic coverage. The map presents the semantics of the topics and their associated keywords as they are used in relationship to each other, essentially encoding the propositions for a given term-by-document. If two keywords are used in the same sentence, they are linked in the map. The thickness of a link indicates the number of sentences that connect two keywords, while the colour of a keyword indicates its dominant topic. Hovering over an edge will reveal the actual sentences where the two keywords have been linked. The map allows a user to consider the comprehensiveness of the essay’s topic coverage. A good map should link a variety of keywords, not just the central word to the others. That is, a good essay should be discussing the topics from as many angles as possible, relating them with other topics, and not simply stating them on their own.

The semantic mapping tool develops a visualisation by analysing a text to reveal its important concepts, relationships and relevant network attributes. Using the capabilities of TML, text pre-processing methods, such as stemming, deletion and generalisation, are applied to condense the data and identify the concepts. The relationships between concepts are established, by identifying linking phrases and assessing the position of the concepts in a sentence or paragraph. This approach gives both meaning and direction to the relationships. The mined information is then further refined by performing a network analysis to measure the relative importance of concepts and relationships. This ability provides a means to constrain the data so that weakest cognitive links disappear, based on the closeness rating. Finally, all this information is then visualised using a map, as illustrated in Figure 4.8.
Figure 4.8: A topic coverage map generated from a short essay. The colour of the nodes and arrows indicates the topic of the associated keywords and their connecting sentences.

The semantic structure of the essay revealed in the map visualises the conceptual relationships that form one’s reasoning and understanding of key concepts. Using this approach, the similarities and differences between concept maps can be clearly visualised, making concept mapping an effective tool to evaluate and compare the knowledge of groups and individuals. In our model, a concept map is expressed using a directed edge-node graph, where the nodes correspond to the concepts being represented and the edges correspond to the relationships between these concepts. Each relationship has properties to define its type (for example, ‘is a’), frequency and direction.
The interface dynamically generates an interactive concept map that the user can explore and manipulate. In order to visualise this information in an informative way, spatial position, colour, size, thickness, and shape have all been used to encode information in the graph visually. Colour-coding has been effectively used to allow the user to differentiate between clusters and the degree of colour shading to indicate the position of a concept in the hierarchical structure of the concept map. Spatial position has been used group nodes into clusters, to distinguish between related and unrelated nodes visually.

In the field of graph visualisation, spring embedding algorithms are typically used to position the nodes of a graph into a layout that satisfies presentation requirements. In spring embedding methods, nodes are seen as physical bodies that cause repelling forces on one another and edges between nodes are seen as springs that cause attraction forces between nodes. The final layout of the graph is a solution in which the forces on each node in the graph are in equilibrium. Importantly, the layout also needs to consider the spatial positioning of the graph clusters in order to visualise the proximity of clusters and related nodes. Our layout algorithm is a spring-embedded method to set the initial spatial position of the nodes in the graph, ensuring adequate space around each node. However, an evident deficiency with this layout approach is that there is a tendency for overlap when displaying a large number of nodes. This results in a decrease in insight into the content of the concept map, which consequently makes it difficult to gain a quick overview. Users can move, resize and change the layout of the graph according to their needs. Users can additionally use the available zooming and panning options to explore the visualisation further.

Support for Collaborative Writing

Although not discussed in this thesis, Glosser has also been designed to support collaborative writing. Glosser uses collaborative writing functions that contain revisions and authors. It is possible to see who contributed which sentence or paragraph as well as who used and contributed to each topic. It is possible to
uncover who linked keywords between sentences, paragraphs and topics. Collaborative writing adds a new dimension when considering coherence, due to the inconsistencies in knowledge and writing styles between authors.

In Gloser, holding the mouse pointer over a paragraph highlights how much each author contributed to it as well as the timestamp for that revision in a tooltip style box. Currently, Gloser uses string matching, searching for the most recent diff in which a sentence appears, in order to find the author of the sentence. Whilst this approach is simplistic and may seem quite erroneous, evidence collected in a study of reviewing practices in academic writing suggests that collaborative groups rarely make changes to parts of a text written by their co-writers (Kim & Eklundh, 2001).

Gloser also includes an additional tool to analyse the participation of individuals in collaborative groups. This tool presents statistics based on the number of revisions by individual authors, as well as a timeline of when these revisions were made. These tools can help a collaborative team understand how each group member is participating in the writing process.
Chapter 5

Mapping the Semantics of Texts

It is a widely held view in academic research that an essay should have a clear and logical flow of ideas, which are inherently linked and structured through paragraphs. This chapter presents the approach for mapping the paragraph semantics of a document’s topics. More specifically, it presents a visualisation to help assess text semantics subjectively, in addition to the various algorithms used to optimise its meaning and layout. The choice of algorithms used in this approach is based on the theoretical analysis discussed in Chapter 2 and Chapter 3, and the experimental results in Chapter 6. The research presented in this chapter has been described in two conference papers, which quantitatively analysed the semantic topic model approach (O’Rourke & Calvo, 2009a) and presented the topic flow visualisation (O’Rourke & Calvo, 2009c).

5.1 Data Analysis

The topic flow mapping approach uses text mining techniques to model the topic mixture of a document’s paragraphs and map them to a two-dimensional space for visual consumption. The automated mapping approach involves performing the following steps:

1. First, a term-by-paragraph matrix is prepared, after stop-words and low frequency words are removed, and stemming is applied.
2. Second, a topic model is created using Non-negative Matrix Factorisation (NMF).

3. Third, the topic model is projected to a two-dimensional space using multidimensional scaling.

4. Finally, a visualisation of the document is produced.

The approach chosen is based on techniques that are well established and proven in the literature as well as the theoretical suitability of these techniques for the application. It is designed to be applied to most problems without parameter tuning, or substantial work on stopword lists.

In the pre-processing step, after a given English document is decomposed into individual paragraphs, a list of 425 words from the Brown stopword list are removed, and stemming is performed using the Porter stemming algorithm (Porter, 1980). A term-frequency vector for each paragraph in the document is then constructed from the terms’ stems.

The core of the mapping approach is in NMF, which is used to model the topic flow of a document’s text passages based on their associated topic mixtures. A separate NMF model is built for each document, using a document’s paragraphs to generate the term-by-paragraph matrix $X$. The matrix elements of the initial term-by-paragraph matrix can be weighted using a number of schemes (i.e., Log-entropy and TF-IDF) (Ricardo & Berthier, 1999). The results in the Chapter 6 are produced using log-entropy, although the same visualisation can be produced with the other approaches. Log-entropy weights a term $i$ by the log of its frequency $f_{ij}$ in a paragraph $j$ offset by the inverse of the entropy of its frequency across all $n$ paragraphs in a document. The formula for calculating the log-entropy weight of a term entry is reproduced below in Equation (5.1). Log-entropy provides a useful weighting scheme for our purposes because it assigns higher weights to terms that appear fewer times in a smaller number of paragraphs. Thus, the scheme emphasises the importance of infrequent terms while also eliminating the ‘noise’ of frequent terms.
\[
x_{ij} = \frac{\log(1 + f_{ij})}{-\sum_k^n \left( \frac{f_{ik}}{\sum_l^n f_{il}} \right) \log \left( \frac{f_{ik}}{\sum_l^n f_{il}} \right)}
\] (5.1)

NMF generates its topic model by decomposing the term-by-paragraph matrix \(X\) into the product of two \(k\)-rank non-negative matrices, \(W\) and \(H\), so that \(X \approx WH\). In our case, \(k\) is considered the number of latent topics in a document. This makes the choice of \(k\) entirely document dependent. Given that \(k\) represents the number of latent topics in a document, \(W\) becomes a term-by-topic matrix, indicating the weighting of each term in a topic, and \(H\) becomes a topic-by-paragraph matrix, indicating the weighting of each topic in a paragraph. The product \(WH\) is called a non-negative matrix factorisation of \(X\), which can be approximated by minimising the squared error of the Frobenius norm (Meyer, 2000) of \(X-WH\). Finding this solution defines the NMF problem, which is mathematically expressed in Equation 5.2.

\[
F(W, H) = \|X - WH\|_F^2
\] (5.2)

A review of algorithms for solving the NMF problem is available in (Berry, Browne, Langville, Pauca, & Plemmons, 2007). In our approach, we use the prototypical multiplicative algorithm developed by Lee and Seung (2001). This NMF algorithm uses an iterative procedure to multiplicatively update the initial values of \(H\) and \(W\) so that the product approaches \(X\). The update rules for \(W\) and \(H\) are defined in Equations 5.3 and 5.4.

The initial values of \(H\) and \(W\) are generated using the Spherical K-Means clustering method (Wild, Curry, & Dougherty, 2004), such that \(H_{ij} > 0\) and \(W_{ij} > 0\). This structured initialisation has the benefit of faster convergence as well as producing more consistent results compared to purely random initialisation methods.
\[ H_{cj} \leftarrow H_{cj} \frac{(W^T V)_{kj}}{(W^T WH)_{kj}} \]  

(5.3)

\[ W_{ic} \leftarrow W_{ic} \frac{(VH^T)_{ic}}{(WHH^T)_{ic}} \]  

(5.4)

Once the NMF model is built, each topic is represented as a vector of its distribution of terms and each paragraph is represented as a vector of its distribution of terms over these topics. Thus producing the topic model from which an analysis of a document’s semantic structure and flow can be performed.

### 5.2 Quantifying Topic Flow

As topic flow between consecutive parts of an essay is generally considered a positive feature, the assumption was made in this thesis that an essay should have a measurable degree of topic flow, and that on average a well written essays should have a higher amount of topic flow than that of a poorly written essay. However, actually quantifying the topic flow in an essay is a difficult task and a methodology for doing so is missing from the literature. While a break in topic flow can sometimes be a good thing, on average it is reasonable to expect that a well written essay’s topic flow should be better than that which would be expected from random chance.

In the context of this thesis, topic flow is quantified as the average amount of semantic overlap between successive sentences or paragraphs in an essay. The Distance Index (DI), defined in Equation (5.5), measures the sum of semantic distances \( \hat{d}_{ij} \) between consecutive pairs of sentences or paragraphs, ‘centred’ and normalised by the average over all the pairs of sentences or paragraphs in a document. These averages are equivalent to distances that would be expected from randomising the order of the paragraphs. A DI value less than or equal to 0 indicates
a random topic flow, while a DI value greater than 0 indicates the presence of topic flow.

\[
DI = 1 - \frac{1}{n} \sum_{i=1}^{n-1} \hat{d}_{ii+1} - \frac{2}{n \sum_{i<j} \hat{d}_{ij}}
\]  

(5.5)

5.3 Topic Flow Map

This thesis proposes the use of visualisation to map the semantic relationships of a document’s paragraphs, in order to interpret the topic model of a piece of academic writing. The topic flow map visualisation is intended to help assess features related to the structure and flow of an essay. A topic flow map contains exactly the same information as its equivalent essay, but presents this information in a new way to put more emphasis on the essay semantics. This essay visualisation approach provides an assessor with an insight into an essay’s semantic features while still maintaining the subjectivity of the assessment process.

In the essay visualisation approach, the distances between paragraphs in the topic model are mapped to a two-dimensional space using a technique called multidimensional scaling. The distances can be calculated in the standard measures (e.g., cosine similarity or Euclidean distance). Using these measures, the distance between any two paragraphs (not only the consecutive ones) is calculated to create a paragraph-by-paragraph triangular distance matrix. Multidimensional scaling uses this distance matrix to produce a two-dimensional representation (Borg & Groenen, 2005). For example, given the distances between all the cities in a country, multidimensional scaling could be used to plot the relative location of each city on a two-dimensional map. Here, the multidimensional scaling transformation is performed using a procedure called iterative majorisation (De Leeuw, 1988). The iterative majorisation algorithm undertakes an iterative, least-squares approach to
multidimensional scaling by attempting to minimise a loss function. The energy function in Equation (5.6) defines the squared relative errors between the paragraph vector distances \( \hat{d}_{ij} \) and their approximated Euclidean distances \( d_{ij} \) in the two-dimensional space.

\[
\sigma_E = \sum_{i<j} \frac{(d_{ij} - \hat{d}_{ij})^2}{\hat{d}_{ij}^2}
\]  

(5.6)

The iterative majorisation algorithm progressively minimises the loss function by creating a new configuration of points until it converges to an optimal arrangement of the paragraphs at its minima. The result is a two-dimensional representation of the paragraphs described in the distance matrix, with the directions of the actual axes being arbitrary.

Figure 5.1 illustrates a ‘map’ of topic flow generated from the paragraphs of a short essay. A node-link diagram is used to represent the topic flow, with the arrows and text labels used to convey the sequence of paragraphs. The paragraph nodes are plotted on a circular grid with the diameter of the grid equal to the maximum possible distance between paragraphs (i.e., no topic overlap). The similarity between paragraphs in the map is solely represented by their relative proximity with the actual orientation of axes having no meaning. This approach permits a characterisation of the local semantics between two paragraphs as well as the global semantics of these paragraphs in the context of the essay as a whole.
Figure 5.1: A map of the topic flow generated from the paragraphs of a short essay.

The five-paragraph essay is a writing paradigm that has been proposed to structure argument in academic writing (Davis & Liss, 2006). Although the use of the five-paragraph essay is a contentious point in the literature, it is used in this thesis solely as an example to demonstrate key concepts due to its simplistic structure. The format of a five-paragraph essay consists of an introduction paragraph, stating the topic thesis and introducing the main supporting subtopics, three body paragraphs, each of which present different supporting subtopics, and a concluding paragraph, which restates the thesis topic and summarises the supporting subtopics. In mapping the topic flow of an ideal five-paragraph essay, it could be expected to resemble a circular layout of sequential paragraphs, indicating a natural change in topic over the
essay, with the introduction and conclusion paragraphs positioned on similar points. In contrast, a poorly structured essay could be expected to have many rough-shifts in topic, with paragraphs positioned somewhat randomly around the map.

Figure 5.2 illustrates the topic flow maps of two short essays that were assessed according to the MASUS procedure. The essay on the left was given a high grade while the essay on the right was given a low grade. Examining these maps in depth, it can be observed that the topic flow of the high grade essay appears to resemble that of an ideal five paragraph essay, while the topic flow of the low grade essay appears somewhat disorganised. In a formative assessment scenario, the rough topic shifts in the low grade essay may be identified as a problem. This could indicate, for example, a lack of information linking two subtopics, which may require the insertion of an additional paragraph or transitional sentence.

Figure 5.2: A map of an essay with a high grade (Left) and an essay with a low grade (Right). Note the low grade essay has many Rough-Shifts in topics while the high grade essay has a more gradual topic flow that starts and finishes on a similar point.

Whilst the scenario discussed above makes sense for the five-paragraph essay, it is important to keep in mind that there are many genres and paradigms for essay writing that do not follow the same strict structure. Further, essay assessment is subjective and it is unlikely that a strictly optimal set of visual heuristics could be
identified. Such work is more in the domain of AES and is not considered here. Instead, the map is designed to capture the topic flow an essay, leaving the interpretations of the maps to the users themselves or to those who require familiarity with the essay, rather than a basis for external judgment.
Chapter 6

Evaluation: Data and Methods

The research presented in this thesis investigates a number of techniques for providing automatic semantic feedback in academic writing. This chapter evaluates two of these techniques. Firstly, the use of matrix factorisation to measure topic flow, and secondly, the topic flow map visualisation, the main and most novel contribution of this work.

The evaluation presents three experiments performed on three essay corpora. The first two experiments aim to validate the topic flow mapping approach quantitatively. The third and final experiment aims to evaluate the usefulness of the topic flow map by measuring if users gain an insight when performing the complex benchmark task of assessing an essay, with the visual aid of a topic flow map.

6.1 Experiment Datasets

The topic flow maps were evaluated using three essay corpora. These corpora include their associated essay grades from which a quality judgement about the essay semantics is inferred.

BAWE Corpus

The British Academic Written English (BAWE) corpus consists of N=2761 documents written for assignments by university students over a four-year period, from 2004 to 2007. The corpus has been made publicly available for research
purposes through the Oxford Text Archive. The documents have been graded as either Merit (60-70%) or Distinction (70-100%), with the exact numeric grades unavailable. The corpus contains documents written in a variety of genres on various topics, with documents having an average of N=2161.08 (SD=906.70) words, N=97.53 (SD=42.39) sentences, N=29.42 (SD=19.45) paragraphs.

MSU Corpus

The MSU corpus contains N=120 essays written for assignments by undergraduate students at Mississippi State University (McNamara, Crossley, & McCarthy, in press). The essays have been graded from 1 to 6. Compared to the BAWE corpus, the essays in MSU corpus are much shorter, with documents having an average of N=726.20 (SD=114.37) words, N=40.03 (SD=8.29) sentences, N=5.55 (SD=1.32) paragraphs.

MASUS Corpus

The MASUS corpus consists of N=43 short essays handwritten in a timed assessment by students at the University of Sydney in 2007. Due to the timed nature of the assessment and the fact that the essays are handwritten, they are often quite erroneous. The corpus was collected for the internal research purposes and cannot be released publicly due to privacy restrictions. The corpus includes its associated numerical essay marks, which were assessed according to the MASUS procedure. Compared to the MSU corpora, the essays in MASUS corpus are slightly shorter in length, with documents having an average of N=445.85 (SD=120.91) words, N=23.30 (SD=5.04) sentences, N=5.40 (SD=1.74) paragraphs.
6.2 Experiment 1: Quantifying Topic Flow

Before getting users to assess the topic flow, there was firstly a need to measure whether the choice of techniques and parameters used in the paragraph mapping approach is in line with the theoretical justifications discussed in the background chapters. The aim of this experiment was to validate quantitatively whether matrix factorisation can be used to analyse topic flow of an essay.

Methodology

As topic flow is generally considered a positive feature of an essay, the assumption was made that these essays do have a measurable degree of topic flow, and that high graded essays have a higher amount of topic flow than that of low grade essays. In order to quantify topic flow the Distance Index (DI) measure in Equation (5.5) has been used.

For each essay, a term-by-sentence and a term-by-paragraph weight matrix was calculated using the log-entropy term weighting scheme (other schemes had similar results). Since different dimensionality reduction techniques may affect the DI measure, this evaluation compares the results of both the Non-negative Matrix Factorisation (NMF) and Singular Value Decomposition (SVD) matrix factorisation methods. The number of dimensions (topics) used for the matrix factorisation algorithms was kept at $k = 5$ throughout the experiment. This parameter was chosen based on experimental experience; however, similar results were achieved with other $k$ values. Determining the number of dimensions is an open question in the literature, and an analysis of this problem is not the scope of this thesis.

The distance between the pairs of sentences and pairs of paragraphs was calculated using the measure of cosine similarity. The DI was used to calculate and compare the difference in topic flow between the sentences and paragraphs for the graded essays subsets produced using the different matrix factorisation methods.

The experiment was repeated across all three of the BAWE, MSU and MASUS corpora. The three essay corpora were each divided into a low grade and a high
grade subset in order to define a quality benchmark on which to critically evaluate the experiment results. The BAWE corpus was divided into a low graded subset consisting of 575 essays graded as merit and a high graded subset consisting of 295 essays graded as distinction. The MSU corpus was divided into a low grade subset consisting of 72 essays graded 1-3 and a high grade subset consisting of 48 essays graded 4-6. Finally, the MASUS corpus was divided into a low grade subset consisting of 18 essays graded 8-13 and a high grade subset consisting of 22 essays graded 14-20.

**Results**

The experiment results for the sentence topic flow and the paragraph topic flow in the BAWE corpus are summarised in Table 6.1. The results show that the distinction essays had a higher average DI compared to that of the merit essays at both the sentence and paragraph level. This result is in agreement with the assumption in this thesis of topic flow and essay quality. On average, there was a higher amount of topic flow between sentences than paragraphs. This was particularly true for the shorter essays, where one would expect the topics to be more concise and confined within paragraph. A graph illustrating the difference in the average sentence DI and the average paragraph DI produced using the NMF and SVD matrix factorisation methods is displayed in Figure 6.1.
Figure 6.1: A comparison the average sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods for the BAWE corpus.

At the paragraph level, the difference in the mean DI between the graded essay subsets calculated using NMF was found to be statistically significant (p < 0.01). Conversely, the difference in DI calculated using SVD did not indicate any statistical significance (p = 0.19). Similarly, at the sentence level this was also the case for NMF and SVD with p-values of 0.08 and 0.29 respectively.

According to the calculated statistical significance, the NMF algorithm was better able to measure a document’s topic flow compared to that of SVD. However, statistical significance does not consider the size of the effect that topic flow actually has on an essay grade. One way of quantifying this is by calculating the ‘effect size’, which, as the name suggests, measures the actual size of the difference between two datasets. At the paragraph level, the Cohen’s effect size of the DI between the two graded essay subsets was calculated to be larger using NMF (0.18) compared to that of SVD (0.06). Similarly, at the sentence level this was also the case with NMF and SVD having effect sizes of 0.10 and 0.04, respectively.
Table 6.1: The mean sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods on the BAWE corpus.

<table>
<thead>
<tr>
<th></th>
<th>Merit</th>
<th>Distinction</th>
<th>p-value</th>
<th>effect size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMF DI.</td>
<td>0.2506 (0.1132)</td>
<td>0.2622 (0.1190)</td>
<td>0.08</td>
<td>0.10</td>
</tr>
<tr>
<td>SVD DI.</td>
<td>0.1382 (0.0925)</td>
<td>0.1418 (0.0953)</td>
<td>0.29</td>
<td>0.04</td>
</tr>
<tr>
<td>Paragraph</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMF DI.</td>
<td>0.1626 (0.1605)</td>
<td>0.1908 (0.1579)</td>
<td>&lt;0.01</td>
<td>0.18</td>
</tr>
<tr>
<td>SVD DI.</td>
<td>0.0866 (0.1570)</td>
<td>0.0964 (0.1526)</td>
<td>0.19</td>
<td>0.06</td>
</tr>
</tbody>
</table>

The same experiment was repeated using the MSU corpus. Although the size and number of essays in the MSU corpus is considerably less than BAWE corpus, it does, however, have a larger range of grades available. The results of the experiment from the MSU corpus are summarised in Table 6.2. The mean sentence DI repeated the results using the BAWE corpus, and the same trend of lower statistical significance and higher effect size was recorded using NMF compared to SVD. However, the average paragraph DI was less present in the MSU corpus, with a value of close 0 and little difference between the graded essay subsets.

Table 6.2: The mean sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods on the MSU corpus.

<table>
<thead>
<tr>
<th></th>
<th>Graded 1-3</th>
<th>Graded 4-6</th>
<th>p-value</th>
<th>Effect size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMF DI.</td>
<td>0.2079 (0.1424)</td>
<td>0.2590 (0.1638)</td>
<td>0.0425</td>
<td>0.3317</td>
</tr>
<tr>
<td>SVD DI.</td>
<td>0.1283 (0.1190)</td>
<td>0.1680 (0.1515)</td>
<td>0.0629</td>
<td>0.2929</td>
</tr>
<tr>
<td>Paragraph</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMF DI.</td>
<td>-0.0107 (0.2097)</td>
<td>0.0349 (0.1852)</td>
<td>0.1217</td>
<td>0.2289</td>
</tr>
<tr>
<td>SVD DI.</td>
<td>0.0376 (0.1780)</td>
<td>0.0196 (0.2220)</td>
<td>0.3077</td>
<td>-0.0904</td>
</tr>
</tbody>
</table>

With such a strong indication of sentence topic flow seen in the MSU corpus and the availability of exact numerical graded, the result was investigated in more detail.
A graph of the mean sentence DI for the exact numerical grades in the MSU corpus is illustrated in Figure 6.2. The graph indicates that on average the sentence DI of the essays increased with the grade.

![Graph showing mean sentence DI vs score](image)

**Figure 6.2:** The mean sentence distance index (DI) for the exact numerical grades in the MSU corpus.

The same experiment was repeated again on the MASUS dataset. The essays in this corpus are smaller in number (N = 43) and length compared to the MSU corpus. The results of the experiment on the MASUS corpus are summarised in Table 6.3. The results are in agreement with those found using the MSU corpus. The essays had a high mean sentence DI, but low mean paragraph DI. This result indicates a strong presence of topic flow between sentences, but not between paragraphs. Similarly the same trend seen in the Bawe and MSU corpora of lower statistical significance and higher effect size for NMF compared to SVD was observed. However, the difference in average DI between the graded essay subsets was not found to be statically significant.
Table 6.3: The mean sentence and paragraph distance indexes (DI) produced using the NMF and SVD matrix factorisation methods on the MASUS corpus.

<table>
<thead>
<tr>
<th></th>
<th>Graded 8 to 13</th>
<th>Graded 14 to 20</th>
<th>p-value</th>
<th>effect size</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sentence</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMF DI. Mean (SD)</td>
<td>0.0894 (0.1112)</td>
<td>0.1267 (0.1525)</td>
<td>0.2076</td>
<td>0.2702</td>
</tr>
<tr>
<td>SVD DI. Mean (SD)</td>
<td>0.0539 (0.1546)</td>
<td>0.0764 (0.1295)</td>
<td>0.3117</td>
<td>0.1633</td>
</tr>
<tr>
<td><strong>Paragraph</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMF DI. Mean (SD)</td>
<td>-0.0153 (0.1900)</td>
<td>0.0573 (0.2015)</td>
<td>0.1301</td>
<td>0.3718</td>
</tr>
<tr>
<td>SVD DI. Mean (SD)</td>
<td>0.0125 (0.2148)</td>
<td>0.0612 (0.1707)</td>
<td>0.2299</td>
<td>0.2453</td>
</tr>
</tbody>
</table>

**Discussion**

The result indicates that NMF and SVD do capture some effect of semantic flow, which is likely due to greater argument overlap by literal repetition of topics from sentence to sentence and paragraph to paragraph. The cosine distance between the paragraphs using NMF was on average less than in SVD. Both these methods indicated that on average they were able to capture the degree to which consecutive sentences and paragraphs discuss similar semantic content.

In general, the effect size of the topic flow on the essay grade was small, but present. Although an effect size of less than 0.20 is often described as small in education research, it can be considered a substantial result in this context. If it could be shown that making a small and inexpensive change would raise academic achievement, then this could be a very significant improvement, particularly if the improvement applied uniformly to all students, and even more so if the effect were cumulative over time.

As previously stated, a break in topic flow can sometimes be a good thing, but on average, it is reasonable to expect that a well written essay’s topic flow to be better than random. In this sense, the DI measure is somewhat *ad hoc* and prone to noise, which is another reason why it had such a small effect size. It should also be noted
that DI did not necessarily always strongly relate to an essay grade. Indeed, the results contained many examples of distinction essays with poor topic flow (i.e., essays that had a worse topic flow than would be expected from random chance). This is partly because there are many factors that influence an essay grade, and thus no single measure can exactly account for a grade alone. Such research is in the domain of automatic essay scoring, for which numerous different scoring metrics exist.

6.3 Experiment 2: Visualising Topic Flow

Experiment 1 demonstrated the effectiveness of matrix factorisation as a technique for measuring the semantic flow of an essay. However, mapping these semantics to a two-dimensional representation presents a different and challenging problem. In order for the resulting visualisation to be meaningful, the Multidimensional Scaling (MDS) projection needs to preserve the relativity of the paragraph distances. The aim of this experiment is to evaluate the accuracy of the MDS paragraph mapping approach and determine the limits to which such an analysis can be considered reliable.

Methodology

In the paragraph mapping approach, the measure of fit for an MDS configuration is defined by the loss function in Equation (5.6). As previously mentioned, some scholars suggest that an MDS configuration with an error of less than 20% can be considered accurate. In some applications, others suggest that the rank order of the distances is more important than the actual magnitudes, and that an MDS configuration can be considered accurate up to a point such that the rank order of distances is no longer be preserved. In evaluating the accuracy of the topic flow maps, this experiment considers both the numerical and ordinal error of the distances in the MDS configuration.
Results

The experiment was performed on the same BAWE corpus essay subset used in Experiment 1. The subset contains 870 essays, each of which consist of essays with between 5 and 80 paragraphs. The experiment was performed multiple times using the Non-negative Matrix Factorisation (NMF) method to created semantic spaces constrained to $k = 5, 10, 15$ and $20$ dimensions for each document. The fit errors of the documents’ MDS configurations were recorded for each of the different $k$ dimensionality constraints.

Figure 6.3 displays a scatterplot of the MDS fit errors versus the number of paragraphs of in each document for the different $k$ dimensionality constraints. For the BAWE document subset, the projected Euclidean distances in the MDS configuration were found to be highly similar to the cosine distance calculated directly from the NMF topic model. As expected, the amount of error in an MDS configuration for a document increased according to the number of paragraphs and the number of dimensions in the semantic space used to model them. At its highest, the fit error reached approximately 0.14 for documents with up to 80 paragraphs. This value is well within the range of which researchers consider a configuration accurate in the MDS literature.

Of course, there are other document dependent factors influencing the error, such as the length of the paragraphs and document vocabulary, which is why essays with the same number of paragraphs can differ so greatly in error. Further, due to the $k<\min(m, n)$ constraint on the dimensions of the NMF model, it can be observed that the scatterplot in Figure 6.3 begins from a common point and diverges only once the $k$ dimension limits of the documents are reached. However, after such a point, the error value of the document for a given $k$ dimensions remained fairly constant despite any subsequent increase in the number of paragraphs.
Figure 6.3: A scatterplot of the fit error in an MDS configuration versus the number of paragraphs in each document for semantic spaces with different $k$ dimension constrains.

Figure 6.4 displays a scatterplot of the Pearson’s correlation between the cosine distances in the NMF model and the projected Euclidean distances in the MDS configuration versus the number of paragraphs in each document for the different $k$ dimensionality constraints. Similar to the results in Figure 6.3, the error in the rank order of the paragraphs distances in an MDS configuration increased according to the number of paragraphs and the number of dimensions in the semantic space. At the worst, the Pearson’s correlation reached a value of approximately 0.58 for documents with up to 80 paragraphs.
Figure 6.4: A scatterplot of the Pearson’s correlation between then distances in the NMF model and the distance in an MDS configuration versus the number of paragraphs in a document for semantic spaces with different $k$ dimension constraints.

Discussion

The experiment results showed that the MDS algorithm was able to accurately project the paragraph distances to a two-dimensional representation for a range of different length and dimensionality constraints. The paragraph mapping approach is reliable in the sense that the distance magnitudes and rank order were largely preserved. Thus, the semantic similarity of a document’s paragraphs, as defined in the NMF model, can be accurately conveyed in two dimensions for visualisation.

The error in representing documents with from 5 to 80 paragraphs was low, provided that a convergence for the MDS configuration could be found. Of course, this is by no means an absolute certainty, and will vary on a document-by-document basis. However, the results give confidence that the MDS projection accurately reflects the paragraph semantic features in the NMF model that the approach attempting to convey.

The paragraph mapping approach was shown to be accurate for visualising essays of up to at least 80 paragraphs; a number far exceeding that which an essay visualisation would become overly cluttered and unreadable. Importantly though,
the size of the documents for which the approach was accurate far exceeds the length of a typical short essay (less than ten paragraphs). This is the approximate length of the essays in the MASUS corpus, which were visualised and evaluated by human assessors in Experiment 3.

### 6.4 Experiment 3: Measuring Visualisation Insight

The aim of this experiment was to evaluate the usefulness of the topic flow maps as a visual aid in the characterisation of texts’ semantic structure and flow.

**Methodology**

In order to evaluate the usefulness of the paragraph mapping approach meaningfully, a methodology is required that addresses the needs of the essay assessment scenario. This experiment therefore focuses on measuring the insight gained through a controlled experiment with a benchmark task of assessing essay structure and development of the text as outlined in the MASUS procedure. The structure and development provides suitable and proven criterion for characterising the semantic structure and flow of the essay.

The experiment was designed to directly measure the usefulness of the topic flow maps and ultimately determine whether they are in fact achieving their purpose. The results were evaluated by measuring the difference between time for participants to complete the benchmark task and inter-rater agreement with an expert MASUS rater, both with and without the visual aid of the topic flow maps. It is hypothesised that the structure and development of an essay can be subjectively assessed faster, more accurately and more consistently with the visual aid of a topic flow map.

The participants were instructed to perform a trial on each document in the MASUS essay corpus. Each trial required the participant to quantify the overall structure and development of the essay as outlined in the MASUS procedure: ‘Do
you start with an introduction? Is there a clear flow of ideas in your essay? Is there a conclusion that follows from the rest of the essay? Do you make paragraph breaks in places that reflect the structure of your essay?’ (see Appendix A).

The process of assessing an essay using the topic flow map is illustrated in Figure 6.5. For each trial, the response of the participant was recorded along with their completion time. The usefulness of the topic flow maps was evaluated by comparing the difference in completion time and inter-rater agreement for the essay subsets with and without the use of the maps.

![Diagram](image)

*Figure 6.5: A diagram illustrating the essay assessment process with the visual aid of an essay’s topic flow map.*

The participants are to perform the experiment using a purpose built essay assessment tool. Figure 6.6 illustrates the main interface of the assessment tool, which consists of an essay side by side with its corresponding topic flow map (if applicable). Each of the essay paragraphs are numbered, so they can easily be identified in the map. The MASUS structure and development criterion is stated at the top for easy reference. Below the MASUS questions is a list box for the participant to record their score. Each time a participant changes the score in the list box, their assessment time is updated and logged by the tool.
### Structure and Development of the Essay

- Do you start with an introduction?
- Is there a clear flow of ideas in your essay?
- Is there a conclusion that follows from the rest of the essay?
- Do you make paragraph breaks in places that reflect the structure of your essay?

<table>
<thead>
<tr>
<th>Essay Subset Preparation</th>
</tr>
</thead>
<tbody>
<tr>
<td>That the use of English is increasing around the world in the past few decades, it is undeniable. However, considering English as a global language has always been a matter of controversy. Some people think it is not an appropriate approach if we consider English as a global language and it can have some negative outcomes. While others believe that using English as a global language is a positive approach. There is no doubt that first the concept of English as a global language will be discussed and then the positive and negative impacts of accepting English as a global language will be outlined. Many people think that the only way of removing the foreign language barrier is to choose a particular language as a global language. There are also some historical examples available on this issue. The Latin was used as a medium of education in Western Europe during the Middle Ages. The problem that might arise at this stage is the communication in different parts of the world. In this respect, however, English because of the high level of international use and high familiarity worldwide is the most important competitor to become a global language. The language of all of the important institutions in the world is English. English is the main language that scientists usually write in. More than 80% of the internet content are in English and English is the main language for technology. All these reasons can make English the most eligible language for being a global language. Accepting English as a global language will divide the defenders of this idea to two groups. One group consider it as a positive development. These people think that English is one of the most eligible languages for being a global language. They think that English is a good language and it can have a positive effect on English and it can cause the development of different versions of English. This is why it can make English mutually understandable. And secondly, there is an aspect of globalization of English is the increasing of internet and socio-economic inequalities. English can be considered as the most eligible language to become a global language. However, it is not too early to consider it as a global language. On the other hand, considering English as a global language can both be a positive or negative development and it can not be only considered as either of those.</td>
</tr>
</tbody>
</table>

---

**Figure 6.6:** The main interface of the essay assessment tool. An essay (Left) is displayed side-by-side with its corresponding map (Right).

In total, at least two participants are required for the experiment to perform the task with and without the visual aid of the topic flow maps. Importantly, the experiment participants are required to be chosen with an emphasis on the domain of academic essay writing in order to provide critical metrics from which the results can be evaluated.

### Essay Subset Preparation

Due to the small size of the available MASUS corpus, the reliability of the experiment results was also highly dependent on the allocation of the essay subsets. As such, it was first necessary to verify the consistency of the generated essay subsets, so as to alleviate any unforeseen bias that may be affected by factors relating to the essay word length, number of paragraphs or score.

Firstly, essays with only one or two paragraphs were excluded from the experiment, due to the lack of any useful semantic information that could be abstracted from such a small number of paragraphs. This left 40 essays for
consideration. The 40 essays were divided into two subsets of 20 essays each. The two subsets were randomly generated from the MASUS dataset until a selection was found with roughly equal distributions of words, paragraphs and scores. A comparison of the differences in the distribution of words, paragraphs and scores of the two MASUS essay subsets is summarised in Table 6.4.

<table>
<thead>
<tr>
<th></th>
<th>Map</th>
<th>No map</th>
</tr>
</thead>
<tbody>
<tr>
<td>Essays</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Words</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean (SD)</td>
<td>455.80 (116.17)</td>
<td>435.85 (127.71)</td>
</tr>
<tr>
<td>Paragraphs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean (SD)</td>
<td>5.60 (1.50)</td>
<td>5.75 (2.05)</td>
</tr>
<tr>
<td>Rating</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean (SD)</td>
<td>3.15 (0.65)</td>
<td>3.20 (0.59)</td>
</tr>
</tbody>
</table>

**Results**

The MASUS essays were each assessed by two postgraduate tutors, one a native English speaker (Rater 1) and the other a non-native English speaker (Rater 2). The results of the time taken for the raters to assess each essay is summarised in Table 6.5. In order to eliminate the effect of essay length, the time taken to assess an essay was normalised to a measure of words per second. On average, both raters assessed the essays faster with the visual aid of the maps than without. However, the difference in the time taken to assess the essays subsets for Rater 2 was not statically significant (p = 0.08), while Rater 1 was (p < 0.01).

It took Rater 2 longer to assess the essays for both datasets compared to Rater 1. The fact that Rater 2 is a non-native English speaker is one plausible reason why it took longer on average to read and assess the essays, and also, why the difference in time taken to assess the two datasets was not found to be statistically significant.
Whilst the time difference between the two datasets was not statistically significant for both raters, it was also not significantly worse. Thus, the maps did not adversely affect the time taken to assess the essays and presented no drawback, in terms of time, for assessing an essay.

Table 6.5: A summary of time taken for the two raters to score the MASUS essays, both with and without the visual aid of the topic flow maps.

<table>
<thead>
<tr>
<th>Rater</th>
<th>Map - Time (words/sec)</th>
<th>No map - Time (words/sec)</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rater 1</td>
<td>4.60 (0.85)</td>
<td>3.24 (0.55)</td>
<td>&lt; 0.01</td>
</tr>
<tr>
<td>Rater 2</td>
<td>4.39 (2.28)</td>
<td>3.64 (0.66)</td>
<td>0.08</td>
</tr>
</tbody>
</table>

The most important dimension for evaluating whether the raters gained an insight into the semantics of the essay using the maps is by measuring how the maps affected the accuracy and consistency of the scores given each rater. This can be measured by calculating the correlation of the rater scores to that of the expert MASUS rater, whom is assumed to be 100 per cent correct. Correlation determines the inter-rater agreement between two raters. In order to obtain a comprehensive view of this dimension of the results, both nominal and ordinal correlation measures are used.

Kappa is a correlation coefficient (denoted by κ), which compares the magnitude of the difference in the values given by the raters, while also taking into account the agreement that could be expected to occur through random chance. Thus, the Kappa coefficient can be used to determine if the inter-rater agreement between the raters exceeds random chance levels, which essentially reflects the accuracy of which a rater used the rating scale correctly. Kappa uses a nominal scale, giving values between -1 and 1, inclusive. Where κ = 1 corresponds to complete agreement, κ = 0 corresponds to a lack of agreement (i.e., the agreement expected from random chance), while a value κ = -1 corresponds to negative agreement.
A summary of the scores and the inter-rater agreement of Rater 1 and Rater 2 with the expert MASUS marker are shown in Table 6.6. The Kappa coefficient of Rater 1 was calculated to be $\kappa = 0.47$ with the visual aid of the maps and $\kappa = 0.07$ without the maps. These values correspond to moderate agreement and slight agreement, respectively, according to the scale given by Landis and Koch (1977). Although this scale is not universally accepted, what can definitely be said is that Rater 1 achieved a higher agreement with the expert MASUS rater with the visual aid of the maps than without.

In contrast, Rater 2 had a Kappa coefficient of $\kappa = 0.07$ with the visual aid of the maps and $\kappa = -0.07$, which indicates a poor agreement and slight agreement respectively. Although the difference in agreement between the two essay subsets was small, it was still present. Therefore, both raters used the rating scale more accurate in the essay subset with the visual aid of the maps.

Whilst the Kappa coefficient indicates a measure of accuracy between raters, it does not necessarily illustrate whether the raters used the rating scale consistently. For example, in scoring an essay, Rater A might assign the scores 1, 2, 1 and Rater B might assign the scores 2, 3, 2. Using the Kappa coefficient, the correlation would be 0, indicating no agreement between the raters even though the raters consistently scored the essays in the same rank order. Looking at the data from a different perspective, the Pearson correlation coefficient (denoted by $r$) is a measure of the linear dependence between two variables. Pearson’s $r$ can be used to measure inter-rater agreement using an ordinal scale, giving a value between $+1$ and $-1$, inclusive. Where $r = 1$ corresponds to complete agreement, $r = -1$ corresponds to complete disagreement

Using Pearson’s $r$, the respective inter-rater agreement with the expert MASUS raters for Rater 1 and Rater 2 was $r = 0.69$ and $r = 0.63$ using the maps compared to $r = -0.14$ and $r = -0.02$ without the maps. In both cases, there was a big different in Pearson’s correlation for the two essay subsets. Therefore, both raters used the rating scale more consistently in the essay subset with the visual aid of the maps.
Table 6.6: A summary of the scores given by the two raters and their correlation with the expert MASUS rater.

<table>
<thead>
<tr>
<th></th>
<th>Score - Map</th>
<th>Score - No map</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Rater 1</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean (SD)</td>
<td>3.25 (0.79)</td>
<td>3.00 (0.92)</td>
</tr>
<tr>
<td>Kappa κ</td>
<td>0.47</td>
<td>-0.07</td>
</tr>
<tr>
<td>Pearson’s r</td>
<td>0.69</td>
<td>-0.14</td>
</tr>
<tr>
<td><strong>Rater 2</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mean (SD)</td>
<td>2.35 (1.09)</td>
<td>2.15 (0.83)</td>
</tr>
<tr>
<td>Kappa κ</td>
<td>0.07</td>
<td>-0.07</td>
</tr>
<tr>
<td>Pearson’s r</td>
<td>0.63</td>
<td>-0.02</td>
</tr>
</tbody>
</table>

**Discussion**

The raters’ assessment of the MASUS Structure and Development of the essays was more accurate and consistent with the visual aid of the maps. Rater 1 used the rating scale significantly more accurately and consistently, with the visual aid of the maps than without. Conversely, Rater 2 was slightly more accurately with the visual aid of the maps, but used the rating scale in a much more consistent way. These results suggest that the raters did gain an insight into the semantic structure and flow of the essays, as modelled using NMF, using the topic flow maps.

While the result do provide evidence to support the hypothesis that an essay’s semantic features can be subjectively assessed faster, more accurately and more consistently with the aid of visualisation. The experiment needs to be repeated with more raters on more datasets before any firm conclusions can be made in relation to the usefulness of the approach. The corpus used in the experiment was small (20 essay with maps and 20 essays without maps) and only two raters participated. This, as in any study where humans are needed, was logistically difficult due to cost and time constraints. However, with such promising results in an exploratory and new idea does give cause for further research on the usefulness of visualisation for interpreting essay semantics.
6.5 Discussion and Future Work

Defining what the semantic space of an individual document means and determining how it can be interpreted to provide automatic feedback is in no way simple. Intuitively, the semantic space of a single document tells us something about the structure and flow of the document’s topic content. How to best quantify this intuition using a single document is still an open research question. What is certain from the literature is that humans need structure and flow in a text for learning to occur and that there must be at least some change in the semantic content across a text’s parts. A break in topic flow is not necessarily good or bad, but there should still be a measurable semantic structure in an essay. Over the course of a text the topic will shift in such that it is likely that any part of a text will be less similar to parts that are more distant.

Visualisation was used as a technique in this thesis for interpreting the semantic structure and flow of an essay for the purpose of providing feedback. The topic flow map was designed as an analysis tool, a way to view the semantics of a document in a different way. The magnitude of the distance between two paragraphs is meaningless on its own. This is one reason why the topic flow map makes sense for subjectively interpreting the semantics of an essay, as it gives meaning to the paragraphs distances by contextualising them relative to all other paragraphs in a document.

The use of visualisation as a form of feedback is quite novel. Thus, the learner experience of using the visualisation as a diagnosis tool in a formative assessment exercise would be largely dependent on the user interface design and the provision of adequate user training. Activities would need to be devised that reflect these needs in the context of the topic flow map visualisation as a guide to identifying potential problems, rather a definitive answer of quality.

In Experiment 1, it was shown that higher quality essays had a higher amount of semantic overlap from paragraph to paragraph and even more so from sentence to sentence. These quality judgements of the essays were based on their respective
essay grades. The results of this experiment demonstrated that on average there is a measurable amount of topic flow in an essay that relates to the quality of an essay. Experiment 2 showed that the MDS algorithm was accurate in mapping the semantic distances between the paragraphs of an essay (up to 80 paragraphs) to two dimensions, thus giving confidence that the information conveyed in the two-dimensional topic flow map of a document accurately represents the expected semantic features.

Experiment 3 reinforced the results in Experiments 1 and 2, indicating that there is indeed something of significance in the semantic structure and flow of an essay. The results of Experiment 3 found that the raters assessed the essays faster, more accurately and more consistently with the visual aid of the topic flow maps than without. The MASUS criteria for assessing the structure and development of the essay required the raters to score the level to which an essay had made proper use of paragraphs to structure the introduction, body, and conclusion of the essay. These aspects were easily identifiable though the maps, solely based on the semantic similarity of the paragraphs’ topic content. These results are in agreement with other work using visualisation to enhance the speed and effectiveness of user tasks. This makes good sense, of course, and the fact that visualisation captures this phenomenon in matrix factorisation may suggest other useful applications for this sort of analysis, as well as possible directions for automatic feedback on essay semantics.

Automatically scoring an essay requires a background corpus of pre-scored essays. This process is corpus dependent, and requires a new model to be built for each new assessment task. The research presented in this thesis used text mining techniques to uncover what a writer has done, but does not attempt make an accurate judgement of whether what they have done is good or bad. This is up to the assessor, but the techniques presented provide the assessor with useful information on which an informed decision can be reached. While it is clear that the semantic space and thus the inter-document distances differ from one essay to the next. Nevertheless, some features do remain constant, as could be concluded from the experiment results. It is plausible that some of these features, such as the magnitude
of a topic shift, could be diagnosed without the use of a background corpus and presented to an assessor for a subjective analysis.

However, as discussed previously, accurately classifying these features is a problematic and challenging task, which can not be assessed on the single dimension of topic flow alone. For example, what exactly does a large or a small shift in topic mean in the context of an essay? Does a text with less semantic overlap between paragraphs indicate less overall coherence or simply that the text covers a more diverse set of topics? Understanding how these types of aspects relate to essay semantics is needed to provide a more objectivity level of feedback. In future work, we hope to evaluate how topic flow as measured by matrix factorisation or other topic models, compared to that of human assessors and how such techniques can be used to provide a more quantitative metric of semantic feedback for diagnosing writing problems.
Chapter 7

Conclusions

The research presented in this thesis analysed the applicability of matrix factorisation methods and document similarity comparison for quantifying the latent semantic features of an essay and evaluated how visualisation can be used to interpret these features. This research was presented and evaluated in the context of providing automatic feedback to support formative assessment in academic writing.

An automatic semantic feedback system called Glosser was developed and presented in this thesis. Glosser applies text mining and linguistic techniques to provide feedback based on the semantic features of a text. Glosser was designed as an analysis tool. It is a way to analyse the semantics of a document in a different way for providing feedback in educational settings.

The use of matrix factorisation for measuring topic flow was evaluated using several corpora of essays written by university students. Two matrix factorisation methods, non-negative matrix factorisation and singular value decomposition, were evaluated with respect to the amount of measurable topic flow according to a defined distance index. The results indicated that non-negative matrix factorisation was significantly more suited to the application of capturing topic flow compared to that of singular value decomposition, but the effect size in relation to grades was small. Matrix factorisation provides a powerful measure for modelling the semantics of a text, which can be accurately mapped to two dimensions using multidimensional scaling and interpreted through visualisation.

A novel visualisation, motivated by the MASUS procedure, for assessing aspects relating to the semantic structure and flow of an essay was presented and evaluated.
The topic flow mapping visualisation method involves a process of non-negative matrix factorisation to uncover topics in an essay, followed by multidimensional scaling, to map the topic closeness of the essay’s paragraphs to a two-dimensional representation. Visualisation was presented as a way to provide an ‘insight’ into the latent semantic features of a text. The main benefit of visualisation being that it is able to highlight existing features for consideration in the essay while still maintaining the subjectivity of the essay assessment process.

The topic flow map visualisation was evaluated in such a way that directly measured the insight gained in a formative assessment scenario. This experiment was carried out using a corpus of short essays written by university students, which had been previously assessed by an expert according to the MASUS procedure. The evaluation presented a realistic formative assessment scenario, which demonstrated the usefulness of the visualisation for assessing the structure and flow of an essay. Critical to this analysis was the ability of the visualisation to illustrate the topic flow of an essay’s semantics as a whole, as opposed to simply its consecutive parts. On average the experiment participants were shown to assess the essays faster and more accurately and consistency with the aid of topic flow map visualisation.
References


Ivanic, R., Clark, R., & Rimmershaw, R. (2000). What am I supposed to make of this?: the messages conveyed to students by tutors’ written comments. In M. R. Lea & B. Stierer (Eds.), *Student Writing in Higher Education: New Contexts* (pp. 47-65). Buckingham: Open University Press.


Minka, T., & La, J. (2002). Expectation-propagation for the generative aspect model.


**Appendix A: MASUS Diagnostic Assessment Sheet**

**KEY TO RATING**

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
<th>A</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>excellent/no problems/accurate/very appropriate</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>3</td>
<td>good/minor problems/mainly accurate/largely appropriate</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2</td>
<td>only fair/some problems/often inaccurate/often inappropriate</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1</td>
<td>poor/major problems/inaccurate/inappropriate</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

**A. INSIGHT AND UNDERSTANDING**

<table>
<thead>
<tr>
<th>Score</th>
<th>Item</th>
<th>A</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Have you answered the questions?</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>3</td>
<td>Do you give evidence for your arguments?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2</td>
<td>Do you evaluate evidence critically?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1</td>
<td>Are your arguments convincing?</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

**B. USE OF INFORMATION**

<table>
<thead>
<tr>
<th>Score</th>
<th>Item</th>
<th>A</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Do you use relevant information?</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>3</td>
<td>Do you avoid irrelevant information?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2</td>
<td>Do you integrate information from the readings into your answer?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1</td>
<td>Is there any plagiarism?</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

**C. STRUCTURE AND DEVELOPMENT OF THE ANSWER**

<table>
<thead>
<tr>
<th>Score</th>
<th>Item</th>
<th>A</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Do you start with an introduction?</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>3</td>
<td>Is there a clear flow of ideas in your essay?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2</td>
<td>Is there a conclusion that follows from the rest of the essay?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1</td>
<td>Do you make paragraph breaks in places that reflect the structure of your essay?</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

**D. CONTROL OF ACADEMIC WRITING STYLE**

<table>
<thead>
<tr>
<th>Score</th>
<th>Item</th>
<th>A</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Do you use have a good academic writing style, in terms of:</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td>sentence structure?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td></td>
<td>vocabulary?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td></td>
<td>linking sentences?</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

**E. CORRECTNESS OF LANGUAGE USE**

<table>
<thead>
<tr>
<th>Score</th>
<th>Item</th>
<th>A</th>
<th>NA</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Do you use correct grammar?</td>
<td>A</td>
<td>NA</td>
</tr>
<tr>
<td>3</td>
<td>Do you use correct vocabulary?</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2</td>
<td>Do you use correct spelling?</td>
<td>•</td>
<td>•</td>
</tr>
</tbody>
</table>

Appendix B: Analysing Semantic Flow in Academic Writing
Appendix C: Visualizing Paragraph Closeness for Academic Writing Support
Appendix D: Visualising Social Networks in

Collaborative Environments