An Exploration of Item Relatedness in a User-Curated Corpus of Digital Cultural Heritage

Alexander Wingard

Dissertation presented for the degree of
Master of Science in Computer Science

School of Informatics
The University of Edinburgh
August 2016

Abstract

The purpose of this MSc dissertation is to test the suitability of similarity algorithms on a corpus of user descriptions of artworks contained within the Artcasting application. The principal focus of this dissertation is the implementation and testing of a number of textual similarity techniques which are applied to a corpus of user generated artwork descriptions. Other similarity techniques are designed and implemented according to the unique nature of data in the corpus. The suitability of algorithms is then evaluated using statistics from the corpus as well as user-tagged data from the application. The impact of information filtering and recommendations in the Artcasting application is also assessed. This project seeks to provide new ways for users to interact with the Artcasting application and increase overall engagement with both the Artcasting application and with works of cultural heritage.

Acknowledgements

I'd like to thank my supervisors Dragan Gasevic and Jen Ross who have provided support and feedback throughout this dissertation process. My friend Sam Cates helped conduct field studies at the Gallery of Modern art and deserves recognition for his help. Finally, friends, family and colleagues have all been helpful during the development of this project and deserve recognition for their continued support.

Contents

List of Figures

Chapter 1

Introduction

1.1 Digital Cultural Heritage

The advent of the digital age has pushed back the frontiers of human interaction with culture. Techniques are constantly being developed and refined in a multitude of fields, leading to this expansion. Detailed data structures allow curators to explore new links between exhibits, modelling techniques allow conservationists to explore all manner of artifacts - from long-extinct languages to far flung archaeological sites - and innovative applications enthrall and inspire museum visitors. All of these exciting developments fall under the umbrella of Digital Cultural Heritage. This is a broad field with many sub-fields and active areas of research, but its principal concern is the use of digital media in the preservation and exploration of items of cultural heritage [2].

One sub-field of digital cultural heritage focuses on engaging visitors to museums with the items of culture on display. The rise in internet usage, along with the proliferation of smartphones has brought about an increase in the number of exhibition spaces using mobile applications to reach visitors. Where previously museums and galleries had websites for providing visitors with opening times and what's on guides, almost every exhibition space is now associated with number of different applications used to reach their visitor base. These applications may be web-based or accessed from a smartphone and their functionality is wide ranging. The services provided by applications range from providing visitors access to information about upcoming exhibitions, to streaming live talks and virtual tours of the gallery [3]. Applications also exist to help improve visitor experience in museums. Whether this is by recommending new exhibits that users might enjoy or by allowing users to plan their visit [4, 5, 6, 3], these applications provide valuable interfaces through which gallery visits can be planned, supported and evaluated.

This dissertation project is concerned with a particular application which encourages and measures visitor engagement with exhibits. The Artcasting application asks users to express their personal feelings about artworks displayed in the gallery and uses this data to evaluate exhibitions. It is described below.

1.2 The Artcasting Application

The Artcasting application is described in detail in the proposal for this dissertation [7]. A description is included here for the reader. The Artcasting application [8] has been developed with Artist Rooms on Tour [9]. The application aims to gather visitor feedback on their experiences in exhibitions in art galleries while encouraging users to think more deeply about artworks. It does this by providing a novel interface through which users can communicate their feelings about artworks seen in the gallery.

Figure 1.1: A screenshot of the Artcasting application showing artcasts sent to North America.

Once in the gallery, users may download the application to their Android or iOS devices and 'cast' artworks from an exhibition to a specific time and place. Users choose a geographical destination and a time to send artworks to, and include an explanation which details the reasons for sending an artwork to a particular location. This collection of temporal, spatial and linguistic data is called an artcast. After it has been created created, an artcast travels to its geographical and temporal location at a speed chosen by the user and can be 're-encountered' once it arrives. Re-encounters occur when application users enter the vicinity of an artcast that has completed its journey. Geo-fencing technology is used to display the re-encountered artcast to users. Artcasts have been sent to a range of destinations around the globe (figure 1.1) and all artcasts are visible from the map view where users are free to browse artcasts from the application. On selecting an artcast, a user is displayed the artwork that has been cast and can choose to see the date and explanation for the artcast (figure 1.2).

Underpinning the philosophy of this application is the new mobilities paradigm. This paradigm in the social sciences explores how groups and individuals are affected by the state of near-constant transit prevalent in modern life [10]. The application seeks to gather data from users on the relationship between their movements and how they feel about artworks.

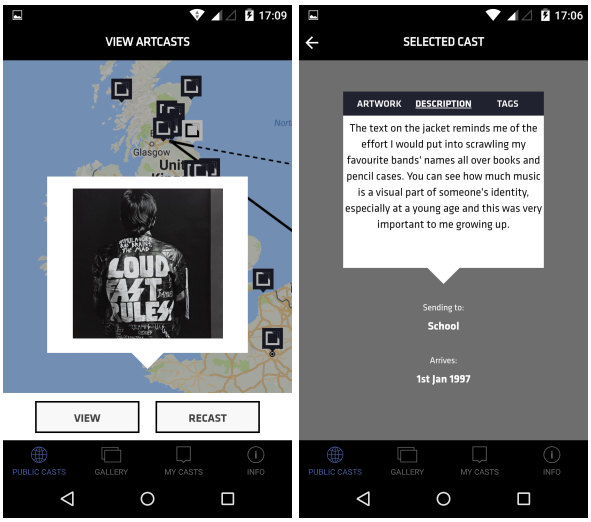

Figure 1.2: Screenshots showing a selected artcast in the map view, and its description.

1.3 Outline

In the following chapter background research and context for this project is given. The Artcasting application is compared to other projects in the field of Digital Cultural Heritage and the foundations of our approach are put in place. Chapter 3 outlines the requirements of a recommender system for the Artcasting application and the research methodology for the project. Chapter 4 discusses the high level design of the system as well as possible technologies for use within system. Chapter 5 discusses specific decisions that were made when implementing the design. Chapter 6 focuses on the evaluation process, using data gathered from user surveys and algorithm performance statistics to evaluate aspects of the project. Finally, we draw conclusions from our work the project and give recommendations for future work.

Chapter 2

Background

2.1 Artcasting in Context

In order to understand the wider aims of the Artcasting project it is useful to examine some other projects which use digital media to increase engagement and interest in cultural heritage. By doing this we gain a deeper understanding the application's functionality, allowing us explore ways in which this functionality can be expanded upon. As the application represents a genuinely new and exciting approach engaging visitors with exhibitions and gathering data on visitor experiences in galleries, it is important to analyse key characteristics of other works to help understand why the application is successful.

Many applications have been created to bring items of cultural heritage to users over the web. One notable and large project is the Art Project [11], powered by Google. This application allows visitors to take virtual tours of a number internationally renowned galleries using Google's street view technology. Artworks are displayed in high quality and users may create their own unique collections by choosing from the repository of artworks available on the application. The application not only allows users to virtually explore exhibition spaces but also to curate and manage their own collections of artworks.

Historypin [12] is an online platform that allows users to upload and share information about history. These items may be histories associated with a particular location, movement or historical period. This project is user driven in that contributors upload individual stories about a subject of their choice and often the stories uploaded come from people who have personal experience of their particular subject but are not necessarily expert historians. In this way the application curates a mixture of primary and secondary source material on aspects of history which may otherwise be forgotten. Users then have the option create personalised tours of this material for themselves or others. Tours can be

virtual or physical. Virtual tours contain pictures and documents which users may browse. Physical tours are designed to be explored in the real world, with data relating to the tour also available through the application. Like the Google Art Project, users are able to explore and create collections of items of cultural heritage, but the user-sourced collection of items adds another dimension to the repository of information.

An application which is somewhat smaller in scope, but equally engaging is Magic Tate Ball [13]. This application uses data taken from a users smartphone (weather, time of day, physical position) to display artworks appropriate for that data, thus providing new ways for application users to view and interact with artworks.

By looking at these works, some key aspects of approaches to digital cultural heritage can be determined. These applications:

- allow access to artworks outside of the gallery space.
- allow users to curate personalised virtual collections.
- curate opinions on items of cultural heritage.
- provide insight into cultural artifacts according to local context.

Artcasting combines these aspects to give unique insight into visitor sentiments associated with specific artworks. By casting art to various different locations users are able to express the influence of their personal experience on their opinions about artworks. As a result, the corpus of artcasts may be thought of as a visitor-curated 'gallery outside a gallery' consisting of unique interpretations of popular artworks.

2.2 Recommender Systems

The aim of recommender systems is to filter information displayed to users presented with large quantities of data which are impossible or impractical to fully explore in normal usage of an application. A recommender system works by gathering data about the user, either explicitly or implicitly, and then working with this data in order to suggest items that may be of interest to them [14].

Recommender systems are most commonly seen in two varieties: collaborative and content-based. Collaborative filtering works by comparing user profiles to predict preferences for a given item. Recommendations are then generated by examining the correlation between profiles. For example, if user A rates items 1 and 2 highly, and user B rates items 1, 2 and 3 highly, user A will be recommended item 3 by a collaborative recommender system comparing the two user profiles. Two well known examples of collaborative recommender systems include the Amazon recommender [15], and lastFM [16].

A content-based recommender compares user profiles to items in order to provide recommendations. Users are asked to provide a rating for items. Similar items to those rated highly are then recommended. For example, in a recommender system for movies, if a user has a tendency to rate comedy movies highly, a content-based recommender system will provide recommendations from the comedy genre. If highly rated movies feature Tom Hanks, it will recommend movies starring Tom Hanks. Popular examples of content-based recommender systems include Rotten Tomatoes [17], Pandora Radio [18] and the Internet Movie Data Base (IMDB) [19].

2.3 Recommender Systems in Digital Cultural Heritage

A number of approaches in the field of digital cultural heritage aim to enhance user experience both inside and outside of the gallery space. A number of successful projects provide personalised tours to gallery visitors [20, 5, 4, 21]. Personalised tours solve issues experienced by visitors to museums. Visitors with specific areas of interest may not always know where to find artworks. Exhibition spaces are often arranged by date (e.g. in a gallery one floor may focus on medieval paintings and another on contemporary works), theme (e.g. in a museum one room is dedicated to reptiles, another to mammals) or some other logical but arbitrary measure. Visitor interests may not align with the layout of an exhibition, so it can be useful to provide personalised results to guide users to exhibits that interest them.

Different approaches to creating these personalised tours are presented. Some focus on the routes taken by visitors through a museum or gallery [5, 21, 6], with a view to enhancing visitor experience by providing them with information about their locale. Generally these systems use wireless tracking technology to develop a model of user preferences. Based on these preferences, users are then presented with information. This information may be descriptive (e.g. explaining artworks they are currently viewing), or prescriptive (e.g. informing users of related works nearby). Using location-based data allows these applications to respond to user actions in real time and does not require the user to enter data in order to generate recommendations. This non-intrusiveness is appealing as it allows users to engage more fully with artworks rather than spending time entering data into an application.

Other systems use recommender systems and information filtering to advise visitors on exhibits they may enjoy [20, 4, 1]. These systems are implemented in a number of different ways. Generally, input is taken from users to create user profiles and a range of methods are applied to compute relatedness between this user profile and exhibits in the collection. Hierarchical arrangements of data are often used to compute similarity between exhibits. Lexical databases such as

Wordnet [22], semantic web technologies [23, 1] and using measures of relatedness implicit in museum curation [1] have all been shown to be successful when applied to recommender systems for cultural heritage. We note that in the field of digital cultural heritage the majority of recommender systems are content-based as this reflects the nature of visitor behaviour in galleries and museums. As visitors tend not to return to galleries frequently it is difficult to build up the detailed user profiles necessary for collaborative filtering. Rather, by asking users to comment on or rate exhibits richer information can be stored about these exhibits, making a content-based recommender system more practical in this setting.

By developing a recommender system for the Artcasting application new ways of 're-encountering' artcasts have been explored. In its current state the application displays artcasts to a user when they reach the physical location an artcast has been sent to on the application. This functionality works well when artcasts are sent to locations easily accessible by users, i.e. in the same town, region or country but renders artcasts sent further afield much less likely to be re-encountered. By introducing a recommender system to the Artcasting application such artcasts are rendered more likely to be viewed by users. Although all artcasts may currently be viewed on the world map on the main screen, the quantity of artcasts is overwhelming. Particular artcasts are not easily identifiable from the map view, and so users know little about the cast before they choose to explore it further. This reduces the chance that a user will find a cast that interests them. By implementing a recommender system novel artcasts that align with user interests are more likely to be discovered.

Chapter 3

Requirements and Methodology

3.1 Requirements

There are a number of characteristics that are desirable for a recommender system for Artcasting. Below is a description of a number of characteristics, some general, others more specific, that are required for this particular recommender system. Many of these criteria are provided in the Recommender Systems Handbook [14].

3.1.1 Accuracy

First and foremost items recommended to users must be in line with user preferences. If a recommender is to improve the application then it must be able to predict user preferences given proper input. Many other desirable characteristics operate in conflict with accuracy, so it is necessary to recognise potential trade-offs to be made when designing our recommender system.

3.1.2 Item Space Coverage

The item space is the set of all items we can possibly recommend to users. A method of providing recommendations should seek to recommend as many suitable items as possible from the item space.

3.1.3 Serendipity

The serendipity of a recommendation tells us to how surprising it is to a user, or how much new information it contains. Providing serendipitous items to a user implies a trade-off with accuracy, as the most serendipitous results are random and therefore inaccurate. As the application has been developed with a spirit of playfulness and creativity at its core we wish to preserve and enhance these qualities through a recommender system. By providing serendipitous recommendations users are excited and surprised, increasing engagement with the application.

3.1.4 Cold Start

Cold start is a well documented problem in recommender systems. For new users and items, ratings are scarce and a recommender may be unable to generate meaningful recommendations. In the context of the Artcasting application this is an important consideration to make, as the application is used most intensively when first installed, then less when the user leaves the gallery. A solution is sought which works as fast as possible for new users and artcasts.

3.2 Analysis

3.2.1 Data Structures

A typical entry in the Artcasting corpus consists of the following: time and date the artcast was created, a title for the artcast, an artwork to cast, a destination latitude and longitude, a user story and, a mark of whether the artcast has been "recast" (currently all values are zero for this aspect of artcasts in our corpus) and a destination date for the artcast. Below is a typical entry for an artcast in the Artcasting corpus:

2015-11-23 22:16:10	maple syrup	molissa_fenley_1983	44.481955	-76.693405	she looks like she would enjoy some sugar, and this is a good place for that.	0	2013	10	1

Each artcast contains a complete set of data. This means all artcasts are comparable.

3.2.2 Approach to Recommendations

As discussed in the previous chapter, a content-based recommender is appropriate for this project. Given that users provide data when submitting an artcast, a content-based approach allows implicit data collection of user preferences from

artcasts submitted by a user. This data can then be used to provide users with recommendations for other artcasts. For this approach to work, each user must create casts from which to implicitly gather data. However, in the application database there are 296 users and only 106 casts. Considering that a number of users post multiple casts, roughly a third of users actively create artcasts. Thus, an implementation of a system based solely on implicit data gathering prevents users who do not create artcasts from accessing recommendations. One option is to add the option for users to rate casts, as in collaborative item to item recommender systems [15]. This builds user profiles for those who do not submit artcasts based on their preferences expressed through rating items.

A rating system for items presents a number of issues. As the items mainly display users' opinions, thoughts and feelings about an artwork it is not always appropriate to provide ratings for these. One other option is to add the now ubiquitous "like" button to our casts in order to provide binary feedback for user preferences. However, the presence of a like button influences the posts a user makes [24]. Artcasting highly values freedom of expression, so this may also be inappropriate. The issue of a rating system is explored further in the design phase.

3.2.3 Time and Place

The current re-encounter functionality of the Artcasting application depends on the time and place and artcast is sent to. A re-encounter depends on two factors: that an app user is in the same location as a particular artcast, and that the artcast has already reached its destination. The idea of time and space is an essential aspect of the application, and a recommender system should include it in some way. Computing temporal and spatial distance or location is an objective measure; the distance between two points given in latitude and longitude is a fixed distance and the amount of time between two dates is a fixed number of days. Assigning each item a fixed score according to these criteria should be fairly straightforward.

3.2.4 Natural Language Processing

A key aspect of a recommender system for Artcasting is Natural Language Processing (NLP). All artcasts contain a title and user story to explain why a user has chosen a particular destination for their artcast. Although this textual information can be viewed by users of the application, and is somewhat secondary to other aspects of the application, in a recommender system it is the focal point of expansion to the application's functionality.

An approach to recommending artcasts according to the natural language associated with them requires document classification and document similarity techniques. The problem of document classification is well documented in NLP

and broadly consists of assigning topics or categories to documents. In this way the content of documents is represented in lower dimensionality space than if the entire document was considered. A plethora of document categorisation techniques are available, and by analysing our corpus work with we hope to assess the suitability of some of these techniques. Document similarity returns a score for documents depending on how similar they are. Similarity can be measured in many different ways. It may be measured according to topics contained in a document, common words with other documents, or the semantic relationships between word senses in documents.

Before discussing NLP techniques in detail, an analysis of the Artcasting corpus of user stories is carried out to develop requirements of a suitable NLP algorithm.

The first thing to be noted about the corpus of user stories in the Artcasting application is its size, and the length of the documents within it. The corpus is comprised of 106 documents at the time of writing. This is a relatively small number of documents compared to other corpora upon which NLP techniques for information filtering have been implemented or designed [25, 26, 27, 28, 4, 29]. Further details are provided by analysing the corpus in a small program. The minimum document length is 2 words, the maximum length 58 words and the average length 13.3 words. The standard deviation from this average is 8.1 words. The average length of a sentence in English is between 15 and 25 words, so an average entry in the Artcasting corpus represents one short sentence.

It is useful to measure the average number of shared words documents per document pair. This gives an idea of how successful statistical models which rely on finding common words, such as TF-IDF [29], LDA [25] and LSA [26] may be when applied to this corpus. An average document pair has 1.2 words in common but after stopword removal (see section 4.3.2), this number is reduced to 0.23. In any implementations of techniques that rely on common words stemming will be used (also see section 4.3.2), which should increase the average number of shared words between documents somewhat. It is important to note that given the standard deviation of document length it is unlikely that these words are distributed evenly throughout the corpus. It is likely that longer documents will contain more common words, and that particular stories will have large numbers of similar documents as they contain contain themes which run throughout the corpus. In this way techniques which rely on finding common words will at least be able to pick up some similarities in the corpus.

From this analysis we develop a number of requirements for an NLP algorithm to be used on the Artcasting corpus of user stories:

- A technique for document classification or similarity must be able to provide recommendations based on our relatively small corpus.
- A technique for document classification or similarity must be able to provide recommendations in the face of short or variable document length.
- A technique for document classification or similarity must be able to func-

tion despite the sparsity of data (common words) often used to classify documents.

Explicit Semantic Analysis

A promising technique for analysing short texts is Explicit Semantic Analysis (ESA) [30, 31, 32, 23]. By taking a few choice samples of text make some inferences can be made about the examples of natural language contained in this corpus. Take the following entries of text from the Artcasting corpus:

1. *"on a billboard along the motorway: on a billboard along a traffic jam. many people will enjoy it"*
2. *"School: The text on the jacket reminds me of the effort I would put into scrawling my favourite bands' names all over books and pencil cases. You can see how much music is a visual part of someone's identity, especially at a young age and this was very important to me growing up."*
3. *"I almost went: like a missed opportunity, almost clear"*
4. *"Nana and Grandpa's house: the photograph took me straight back to my grandpa who died over 10 years ago due to Parkinson's disease. something about the wrinkles, the grey hair and the warm face took me straight to their home in the fens"*

These entries exemplify the varying document length and scarcity of common words in corpus documents, as well as different ways in which users input text into the application. Examples 2 and 4 are written in full sentences, with clear separation between the title and body of text. By reading the text it is fairly apparent what the users are discussing, and the documents can be understood without the image of the artwork as context. In examples 1 and 3 the meaning is less clear and is harder to discern without the context of artwork, time or place. However, studies have shown that ESA can be successful even when documents lack clear structure [31, 30].

3.2.5 Hypotheses

Having discussed and justified a number of possible approaches to implementing a recommender system, hypotheses which will be tested by possible implementations of a recommender system for the Artcasting application can be outlined.

1. For the Artcasting corpus, techniques for document similarity which use explicit semantic analysis techniques give recommendations that are closer to actual user preferences than statistical techniques.
2. Implementing a recommender system in the Artcasting application expands the current functionality and contributes to user experience.

3.3 Methodology

The research conducted must directly address these hypotheses. By analysing the approach taken to evaluation of NLP techniques for document classification a methodology to be used to evaluate this project is developed.

To conduct a survey of the thousands NLP techniques and their evaluations is beyond the scope of this project, instead an analysis of research conducted in a few studies outlines some of the evaluation techniques to be used in this project. Techniques for evaluation of NLP techniques for document classification and similarity fall into two broad categories: those based on statistical analyses [25, 26, 29] and those which use human-tagged data as a gold standard [31, 30, 33, 28, 23]. Understandably, those techniques which focus on a statistical analysis of documents are evaluated statistically, and those which focus on ESA are generally evaluated using human-generated data as a gold standard. Humans have an innate ability to recognise semantic properties of text, and so provide an ideal benchmark when evaluating ESA techniques.

As a result of this difference in approach to evaluation, quantitative data available on the performance of NLP techniques is not always directly comparable. Statistical techniques tend to be evaluated on large corpora, sometimes of tens of thousands of documents. ESA techniques usually focus on smaller corpora, sometimes numbering of just a few tens of documents. The Artcasting corpus has particular characteristics that affect the implementation of any approach, so to ascertain which NLP techniques perform best on this corpus a unique study into the effectiveness of NLP techniques is developed.

3.3.1 Quantitative Goals

In order to evaluate the suitability of NLP algorithms, quantitative data was gathered on both algorithms and user perceptions of text similarity in order to compare the two. By generalising the trends of NLP algorithms on our corpus, key data points for use in the evaluation were extracted. These data points formed the basis of user-driven evaluation of algorithms. In this way direct comparisons between data from users and algorithms was made.

3.3.2 Qualitative Goals

As this project aims to encourage engagement and interest in the application, qualitative data will be essential to determining the impact of a recommender system. This data should relate to how users interact with and react to text similarity, one of the main techniques used in this recommender system. There are a number of techniques for gathering such data, such as interviews, questionnaires

and observations. Time and development constraints will be the determining factor for how this research is carried out.

3.3.3 Strategy

The strategy for collecting our data is dependent on certain aspects of our project. In the limited time available to complete the project, research concerns were balanced with those of development and data analysis. A research method is sought which can be conducted quickly and efficiently while still providing the necessary data for evaluation. With this in mind, a questionnaire on text similarity was chosen as the method of conducting research. Questionnaires represent a time-efficient, tried and tested technique for gathering easily analysable data. There are three main requirements for a questionnaire:

- Gather quantitative data on document similarity to compare to the results of our algorithms.
- Gather quantitative data on user interest in items returned by a recommender system.
- Gather qualitative feedback on the criteria which users believe documents to be similar.
- Gather qualitative feedback on user interest in items returned by a recommender system.

3.3.4 Participants

Potential users of the application were sought as respondents to the questionnaire. Permission was granted to conduct user studies at the Scottish Gallery of Modern Art. A previous deployment of the Artcasting application has taken place here, so this location provided us with direct access to the potential user base of the Artcasting application. Conducting studies here adds further requirements to the questionnaire.

- Questionnaires must be quick to explain and easy to understand, so as to ensure accurate responses.
- Questionnaires must be reasonably quick to complete, so as to minimise impact on respondents gallery visits and maximise the number of questionnaires that can be filled out in a given time.

Number of Participants

As we seek quantitative data from our questionnaires, the more participants we can source the better. Reasonable results have previously been attained using

13-15 participants per text comparison [23], so this figure is used as a baseline.

3.3.5 Data Sets

It is important to discuss data used for evaluation of the recommender system. Resnik and Lin [34] provide good advice when deciding which data to select for evaluation of NLP techniques. One key point is the necessity for separation of training data, development testing or 'devtest' data (data used for formative evaluation during development) and testing data.

Training Data

This is the data used to train models which are used to generate recommendations. Some techniques, such as TF-IDF and LDA require training data in order to generate the models upon which they generate recommendations, although training data may also refer to data used while developing the functionality of an algorithm. It is recommended to keep training data completely separate from testing data, so as to ensure that the test set does not influence the algorithm under test. Depending on which algorithms are implemented, large amounts of training data may or may not be necessary. Where training data was necessary for this project, it is discussed in the appendices of this dissertation.

Development Testing Data

This is the data used during formative testing and evaluation of algorithms. 10 individual user stories (representing about 10% of the data) were selected for preliminary testing of algorithms during the development phase. These user stories contain a few common words and semantically related concepts. These entries were excluded from the test set, as the relationships between these items was well known across all algorithms and would have influenced test results.

Testing Data

The final data set considered is testing data. In this project this is the data presented to potential users to gain feedback on NLP algorithms. Data may be drawn from around 90% of the corpus (excluding training data for certain algorithms) and it is important that choose appropriate data points are chosen. User evaluation indicates the suitability of a single algorithm out of a number of possibilities, so the data points chosen must be comparable to recommendations generated by all candidate algorithms. With this in mind an analysis our data set is carried out in order to select key points for user studies.

As there are just over 90 entries in the corpus available for use in our test set, there are over 4095 potential comparisons to be made between individual texts. When testing 4 algorithms, the number of possible individual computations of document similarity rises to over 16380. A key issue in conducting user studies is selecting a reasonable number of items for users to evaluate while also providing meaningful feedback on the performance of potential algorithms.

In order to do this experiments were run to discover the average similarity of each text to all other texts. This is done for each algorithm under test. An average is taken of the similarity across all algorithms. Documents were then ranked according to this average similarity value to deduce which texts in the corpus were most and least likely to be similar to others across all implemented algorithms.

A single text was chosen from each percentile of this ranked data to analyse in detail, called *percentile texts*. For each percentile text four texts were selected to compare them to. These four texts were chosen from each quartile of the data, so for each percentile text there is one text of low average similarity, one of low-medium, one of medium-high and one of high average similarity. These are called *quartile texts*. This allows us to cover key data points with without comparing an unnecessarily high number of data points. In the user studies, each percentile text is then compared to each of these four quartile texts.

For each percentile text the four quartile texts are different. This was done to ensure that our algorithms could generate some similarity scores for each comparison. Where possible, texts which provided at least some similarity score were chosen for the user studies, to ensure the data was comparable to the implemented algorithms. This did not guarantee all comparisons had comparable data from all algorithms (see section 6.5.2), but it went some way to ensuring that the majority of our data points provided feedback on the performance of implemented algorithms. It is important to note that some unconscious bias may have been introduced when selecting the documents to be compared.

3.3.6 Questionnaire Design

The questionnaire was designed to give feedback on the performance of our algorithms. Potential users of our recommender system were asked to rate the similarity between documents in the Artcasting corpus, and to provide qualitative feedback on their views of these text similarities. Questions were deliberately left open ended and users were not given criteria on which to assess the similarity of texts (unless they were having issues understanding the questionnaire). This was done to help evaluate the suitability of techniques for users in the wild. In the wild users do think of recommendations in terms of document similarity but rather whether overall themes contained in the text are relevant to their selections of artcasts.

Qualitative feedback was also gathered by asking an open ended question on

interest in the similarity. If users expressed interest in a particular text pairing they were asked to provide feedback on why the similarity was interesting to them. This provided a basis for assessment of the impact of introducing a recommender system to the Artcasting application.

The questionnaire and associated ethics form are included in the appendix of this dissertation.

3.4 Evaluation Techniques

There are many different aspects of a recommender system that can be evaluated. These depend on differing user needs in differing situations. The approach to evaluation of some of our requirements is discussed in chapter 8 of the Recommender Systems Handbook [14] and a discussion of how these evaluation techniques may be of use is included here.

3.4.1 Accuracy

One of the most obvious requirements for our recommender system is that it returns items to the user which align with actual user preferences. Root Mean Squared Error (RMSE) is a technique often used to calculate the accuracy of an algorithm. It is given:

$$RMSE = \sqrt{\frac{1}{|\tau|} \sum_{(i,j) \in \tau} (\hat{r}_{i,j} - r_{i,j})^2} \tag{3.1}$$

For a test set τ, (i, j) user-item or item-item pairs, system-generated ratings $\hat{r}$ and user-generated ratings r.

Correlation techniques, such as the Spearman or Pearson correlation coefficients may also be used for vectors of normalised values. For each value in a vector, we compute the norm using the following formula:

$$norm(x) = \frac{x - \bar{x}}{s} \tag{3.2}$$

Where $\bar{x}$ is the sample mean and s is the standard deviation for the vector.

The Pearson correlation coefficient may be used, given:

$$cor_p(A, B) = \frac{1}{n-1} \sum_{i=1}^{n} a_i b_i \tag{3.3}$$

This measures the extent to which a correlation between two variables may be described as a linear function.

Spearman's rank correlation coefficient is another popular measure. It is defined as the Pearson correlation between ranked variables, and is given:

$$cor_s(A, B) = 1 - \frac{6 \sum_{i=1}^{n} d_i^2}{n(n^2 - 1)} \tag{3.4}$$

where d_i is the difference in the rank of the two variables, given $r(a_i) - r(b_i)$. This measures the monotonicity of the correlation between two variables.

3.4.2 Item Space Coverage

Item space coverage can be directly calculated by computing the percentage of items that can be recommended by a system. Other techniques for computing item space coverage are available [14] and these are discussed in greater detail in the chapter on evaluation.

3.4.3 Serendipity

Methods of calculating serendipity of results usually involve using a distance metric for computing the amount of new information contained in recommendations [14]. Again, this is discussed further in later chapters.

Chapter 4

Design and Technologies

4.1 Design Goals

In this chapter the high level design decisions are discussed. Possible technologies and approaches to our recommender system are also discussed. Where appropriate references to other examples of the use of these technologies in other relevant projects are included.

4.2 A Suitable Framework

Before the issues of processing data and classifying artcasts are discussed it is important to have a framework in place around which various feature processing techniques can be constructed. As discussed in requirements, several constraints are placed on a recommender system for Artcasting.

The problem of cold start is exacerbated by the situation in which the application is used. The Artcasting application is used most intensively in the gallery when it is first installed. Once a user has viewed some artcasts and created some their own the application is used less often. There are also a number of users who do not create artcasts, and will only use the application to view artcasts. For these users a rapid solution to the problem of cold start is required in order to engage them as quickly as possible in a limited time frame.

Item space coverage is another constraint to be explored. As the application is designed to help users explore alternative and interesting interactions with art we seek to develop a system which allows users to explore the item space fully. By providing personalised results we may reduce the number of artcasts that a user has access to, as recommendations can only be based on those artcasts that a user has indicated a preference for. Due to the amount of information

contained within the application it is unlikely a user will be able to manually indicate their preference for each artcasts in the application. It is important therefore to provide a way for users to access the greatest number of artcasts possible.

Implementing user profiles adds further issues to a system. As discussed in requirements, implicitly gathering data from a user's artcasts prevents those users who do not create artcasts from accessing the functionality of a recommender system. As a result, an implementation involving user profiles necessitates the development of a rating system for artcasts, the difficulties of which are described in the previous chapter.

Finally, the limited amount of time for this project places constraints the depth of the approach we can take. A trade-off must be made between different aspects of our system in order to develop an effective system. If we wish to fully explore approaches to NLP then we must scale back other aspects of the recommender system.

With these issues in mind we reconsider our approach. The system developed uses a content-based approach, but with certain modifications. Developing an item-to-item recommender system (or information filtering system) provides valuable added functionality to the Artcasting application which is accessible to all users. Although the personalisation of results is not possible in such a system, users may explore the item space more fully. Furthermore, this approach eliminates problem of cold start as recommendations can be made without gathering ratings for artcasts. Users are not tied down to a profile and if they see something that interests them in application, they can find out more about it. Work has posited that for a truly personalised experience this flexibility and ability to deviate from explicit user preferences are essential [1]. Rather than prescribing exactly what users can see, an information filtering system (as opposed to a recommender system) provides users with a guide to related casts rather than limiting them to a precise set of artcasts.

A good information filtering system is essential to implementing a successful content-based recommender [1]. The system developed here provides a template which can be adapted to provide the basis for user profile support should this be desired in future. Importantly, by implementing a more basic recommender more time is spent researching and developing appropriate algorithms for use in our system, which is important given the limited time frame of this project.

4.2.1 Modularity and Structure

For development and testing purposes a modular structure for the system is required. The required functionality of the principal modules of the system are outlined here. Individual modules are required to:

- Parse and store data from the Artcasting corpus.

- Process the destination latitude and longitude co-ordinates of each artcast.
- Process the destination date of each artcast.
- Process the natural language associated with each artcast.
- Rank the data according to the above criteria to provide recommendations.

4.2.2 Correlation and Similarity

Once we have represented aspects of an item (geographical and temporal location, similarity to texts) as a single value, we can then represent each item as a vector of these scores (called feature vectors). Similarity between items may then be computed in a number of different ways. Common measures include the euclidean distance, given:

$$dist(A, B) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \tag{4.1}$$

Where a and b are the components of vectors A and B, both of size n.

Cosine similarity is another frequently used similarity measure, given:

$$sim(A, B) = cos(\theta) = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} \times \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{4.2}$$

This measure returns a value $v \in [0, 1]$ in a positive vector space, with 0 indicating complete dissimilarity, and 1 indicating complete similarity.

The Pearson and Spearman correlation measures discussed in the methodology of this project may also be used to compute similarity between artcasts (see equations 3.3 and 3.4).

A number of techniques for clustering are also available. Although these may be suitable for use in further developments of a recommender system for Artcasting, simple similarity techniques were chosen for this project in order to allow us to focus on other aspects of the system.

4.3 Technologies

As touched upon in previous chapters, a wide range of technologies are available to us when implementing our recommender system. Here key technologies are described, and their suitability for a recommender system for Artcasting is discussed.

4.3.1 Formulae

First, formulae which are used throughout the system are discussed. One key formula is the haversine formula. This formula allows us to compute the distance between two points on the earth's surface given in latitude and longitude. The haversine formula is given:

$$dist = 2rsin^{-1}(\sqrt{sin^2\frac{(lat_1 - lat_2)}{2} + cos(lat_1)cos(lat_2)sin^2(\frac{lon_1 - lon_2}{2}}) \tag{4.3}$$

for r the radius of the earth and lat_i, lon_i the latitudinal and longitudinal co-ordinates of point i.

Most high-level programming languages have good support for computation of a period between two dates. As the most precise data for dates in the Artcasting application is days, the number of days between two dates will provide the distance measure for temporal similarity.

As values for geographical and temporal distance are distance measures distance they may need to be converted into a similarity measure. They can be converted to similarity measures using the following formula:

$$sim(a, b) = \frac{1}{1 + dist(a, b)} \tag{4.4}$$

where $sim(a, b)$ is the similarity between a and b, and $dist(a, b)$ is the distance between a and b.

Another important aspect of a recommender to consider is that of normalisation. Often, it is useful to have directly comparable values for data. In order to normalise the a value v in a vector V we compute:

$$norm(v) = \frac{v}{max(V)}$$

for each value v in vector V. If all values in V are normalised, a vector in which each entry is in the range $[0, 1]$ is returned.

4.3.2 Natural Language Processing

TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) is a commonly used technique for computing document weights. It was first presented by Salton and Buckley [29] and has since been widely applied in NLP. To compute the TF-IDF similarity between a pair of documents the frequency of a given term (or word) in a document is computed. This is the term frequency, tf. The

number of documents containing that term, called the document frequency *df* is also computed and the TF-IDF value is given:

$$\frac{tf}{df}$$

for each term in a document. By computing TF-IDF weights for all terms in a corpus, each document can be represented as a sparse vector of term weights. Given two documents represented as TF-IDF vectors, the cosine similarity measure can be used to give the similarity between the two documents (4.2).

Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is another frequently used technique in NLP [25]. It uses a statistical model to classify documents according to their constituent topics. Training data is input into the model, and the LDA algorithm identifies the latent topics in this training corpus. Once a model has been trained it can then identify topics present in a document, and represent this document as a probability distribution of its latent topics.

Stopwords, Stemming and Tokenisation

For both TF-IDF and LDA, it is important to clean the data upon which we base our models. There are a few steps to be carried out to clean the data from the Artcasting corpus. First we remove all punctuation from our texts and ensure all characters are lower case. Next, we tokenise the words, representing each document as a bag of words. Next, stopwords are removed. Stopwords are the most common words in a language (the, or, and etc.) and removing them focuses an algorithm on the most important words in documents and corpora. Finally, the words are stemmed in order ensure families of words with similar meanings contribute to the same weights where possible (for example, liked and liking have the same stem, lik).

Wordnet

Wordnet [22] is a lexical database containing thousands of word senses of English words. Each word sense is represented by a synset. A synset is a set of words which share a meaning, for example the words “change”, “modify” and “alter” make up a synset in Wordnet. Words may be polysemous (have multiple meanings) and in this case they belong to multiple synsets. Synsets are arranged in a hierarchy according to their semantic properties, with more general terms appearing at higher levels of the ontology, and more specific terms appearing further down.

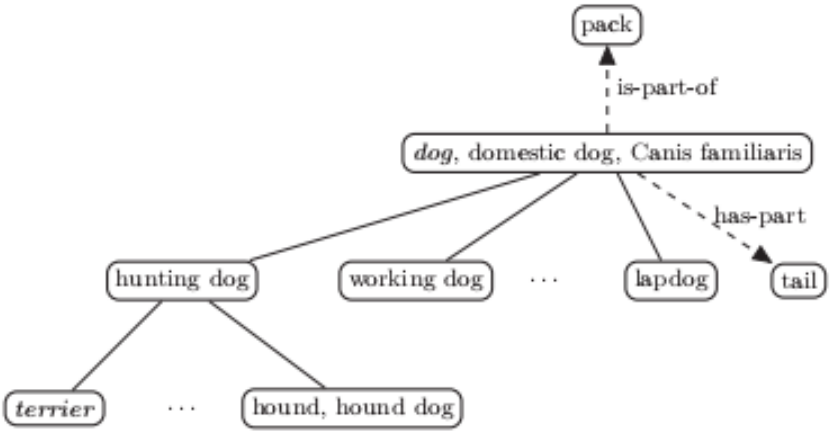

Figure 4.1: A section of the Wordnet ontology: solid lines between nodes represent is-a (hypernym/hyponym) relationships, and dashed lines represent is/has-part (meronym/holonym) relationships. Each node in the graph represents a single Wordnet synset. Taken from Grieser et al. (2011) [1].

Depending on the part of speech (POS) (verb, noun, adjective etc.) of a synset different semantic relationships may be expressed. As the POS is key to the semantic relationships that may be expressed in Wordnet, the database may be thought of as four sub-databases (one for each of noun, verb, adjective and adverb) with some relationships between between them for individual word senses. For each POS different semantic relationships may be expressed. Nouns and verbs have the richest semantic data, with antonymy (possessing opposite meanings, e.g. push is an antonym of pull), hypernymy (possessing a broader meaning, e.g. temple is a hypernym of church) and hyponymy (posessing a narrower meaning, e.g. rat is a hyponym of rodent) all expressed in Wordnet for both. Adjectives and adverbs hold less semantic data. In this project Wordnet 3.0 was used.

Word Sense Disambiguation

Word Sense Disambiguation (WSD) is a technique that determines the sense of a word in context. An algorithm that performs WSD should be able, to use he example given in the title of Lesk's seminal work on WSD, to distinguish between a pine cone and an ice cream cone [33]. It does this by finding the appropriate word sense for the word "cone" using possible word senses from the context words around the target word. There have been a number of improvements to Lesk's original algorithm, notably by Banerjee and Pedersen [35]. If Wordnet-based similarity techniques are to be used to compute document similarity for

our recommender system, WSD techniques will be useful, if not essential, in order to disambiguate word senses expressed in user stories.

The Original Lesk Algorithm

The original Lesk algorithm was developed by Michael Lesk in 1986 as a response to the lack of WSD techniques for Information Retrieval systems [33]. The algorithm seeks to disambiguate the sense of a word in context. Given a word we wish to disambiguate the sense of, we take the set of dictionary definitions (called glosses in Wordnet) associated with that word and compare these to the sets of dictionary definitions associated with every other word in a context window. The dictionary definition which shares the largest number of words (or has the greatest overlap) with other dictionary definitions in the context windows is taken to represent the correct word sense.

Further Developments in WSD

Given the under-performance of the original Lesk algorithm we now examine improvements to the Lesk algorithm. Banerjee and Pedersen present an adapted Lesk algorithm [35] which leverages the structure of the Wordnet database to perform WSD. For a given sequence of words in a document, or window, the algorithm scores combinations of synsets based on the overlap of their glosses, much like in the original Lesk algorithm. However, the algorithm exploits the structure of Wordnet to compare the glosses of a number of semantic relatives of context words (synsets, hpyernyms, hyponyms, holonyms, meronyms, troponyms and attributes). In this way potential senses of the target word are compared to many more glosses, increasing the chance of an accurate disambiguation.

Semantic Similarity

Given two word senses represented by synsets we can compute their similarity by exploiting the structure of the Wordnet database. Given that all variants of the Lesk algorithm discussed here are available for use with Wordnet and represent word senses using synsets we are able to implement these techniques.

Wordnet Hierarchy Methods

There exist many methods for computing relatedness between items in an ontology. Methods have been developed with Wordnet in mind and we summarise a few of them here. Hierarchy methods calculate similarity according the edges between nodes in the Wordnet ontology.

Perhaps the simplest method of computing the similarity between two word sense is the path similarity. In order to compute the path similarity between two word senses or synsets in Wordnet we first find the shortest path (or distance) between two nodes in the ontology, p. The path similarity is then given using the distance conversion given in equation 4.1. In the example given in figure 4.3.2 the path similarity between *terrier* and *dog* is 0.5.

The Leacock-Chodorow similarity measure [36] uses the path similarity, along with the taxonomy depth of synsets to compute semantic similarity. The algorithm first computes the maximum depth of the taxonomy, d, in which the synsets occur, and gives similarity as

$$sim_{lch}(s_1, s_2) = log\frac{p}{2d} \tag{4.5}$$

for p path length between synsets s_1 and s_2.

Information Content Methods

The following algorithms use the Information Content (IC) of word senses in order content in order to compute similarity. The IC of a word sense is formally defined as the logarithm of the probability of encountering the word in a given corpus. These methods use information contained in the nodes, rather than the edges, of the Wordnet hierarchy in order to measure semantic similarity. For these methods we also define the Least Common Subsumer (LCS) of two synsets as the deepest (or most specific) node on a path between both synsets. In the example in figure 4.3.2, the LCS of *hunting dog* and *working dog* is *dog*.

The Resnik similarity of two word senses is defined as the IC of their LCS [37].

The Jiang-Conrath similarity measure [38] uses the IC of the two synsets to be compared and that of their LCS. It is given:

$$sim_{jcn}(s_1, s_2) = \frac{1}{IC(s_1) + IC(s_2) - 2IC(lcs)} \tag{4.6}$$

for the IC of word sense $IC(s)$, synsets to be compared $s1$, $s2$ and LCS lcs.

The Lin similarity [39] computes similarity as:

$$sim_{lin}(s_1, s_2) = \frac{2IC(lcs)}{IC(s1) + IC(s2)}. \tag{4.7}$$

The decision of which similarity measures to implement is made in the following chapter.

Beyond Semantic Similarity

As the average length of a text in the Artcasting corpus is just 13.3 words, a technique for computing the similarity of short sentences is useful. Such a technique is proposed by Li et al. [31]. This algorithm has been designed to retrieve images according to captions, so may be useful for use in the Artcasting application.

Like other techniques discussed here this method uses Wordnet to compute semantic similarity between items in the sentence. An edge-based approach to semantic similarity is adopted. As in other semantic similarity techniques, the depth of synsets in the Wordnet hierarchy are taken into account, with the more specific words further down the hierarchy given greater weight than those at the top. The key difference between this algorithm and other techniques for computing semantic similarity is that it accounts for the word order of sentences.

The algorithm represents documents as both semantic and word order vectors. Semantic vectors contain the semantic information of both documents to be compared. Both a common word set of the two documents and a vector representing each document to be compared are computed. For example, for the documents:

1. Children of all ages like jam.
2. Jam is bad for the health of children.

The common word set is given:

$$W = \{children, of, all, ages, like, jam, is, bad, for, the, health\}$$

and the document vectors are given:

$$W_1 = \{children, of, all, ages, like, jam\}$$

and

$$W_2 = \{jam, is, bad, for, the, health, of, children\}.$$

Semantic vectors for each document are formed by computing the similarity of each word in W_1 to each word in W_2 and vice-versa. Similarity is computed using the shortest path between two synsets and the depth of their LCS in the Wordnet ontology. The dimensionality of the semantic vectors is equal to the cardinality of the joint word set. Given these two semantic vectors, the semantic similarity between both vectors is computed using the cosine similarity.

Word order vectors are computed by assigning each word in the a unique index number. In the above example we have word order vectors 0_1 and 0_2:

$$O_1 = \{1, 2, 3, 4, 5, 6, 0, 0\}$$

and

$$O_2 = \{6, 7, 8, 9, 10, 11, 2, 1\}.$$

We note each word order vector has the same dimensionality is as the common word set, and value of 0 is added to the vector if a word is not present. Word order similarity between two sentences is then computed:

$$S = 1 - \frac{\|0_1 - 0_2\|}{\|0_1 + 0_2\|}.$$

Similarity between two sentences is then computed as:

$$sim(D_1, D_2) = \delta \frac{S_1 \cdot S_2}{\|S_1\| \cdot \|S_2\|} + (1 - \delta) \frac{\|0_1 - 0_2\|}{\|0_1 + 0_2\|}$$

for documents D_1, D_2, semantic vectors S_1, S_2, word order vectors O_1, 0_2 and $\delta \in (0, 1]$.

It is important to note that WSD is not performed in this algorithm, which may affect its accuracy. Despite this, promising results have still been achieved.

This algorithm may be appropriate for a number of documents in the Artcasting corpus which share words and have common word order. For example the pairs:

- *"maple syrup: she looks like she would enjoy some sugar, and this is a good place for that."* and *"storytelling centre: she looks like she has a story to tell!"*
- *"Holiday: Chilling out in the sun"* and *"central park: chilling out in the park"*

contain phrases with the same word order. The implementation of this algorithm will be able to assign similarity to documents which share such phrases.

Chapter 5

Implementation

5.1 Data Structures

It is useful to store the data from the application locally for development and testing. The data from the Artcasting online database was scraped and stored in a JavaScript Object Notation (JSON) file using the Beautiful Soup library for the Python programming language [40].

5.2 Fundamental Decisions

When developing the framework around which all modules of the system are constructed, the first key decision was which programming language to use. The Java programming language (specifically Java 8) was chosen for a number of reasons. Firstly, it provides good support for a range of features we wish to implement, notably time and date processing. There is good technical support and IDEs. It can be used on Android smartphones and implemented for future deployment. Finally, it is the author's most fluent language and felt like a natural starting point.

Once the data had been stored locally, basic modules to parse and store data within the program were developed.

In order to provide recommendations when a particular artcast is chosen by users, we compare the chosen cast (or target cast) to the other casts in the corpus. Each cast is represented by a vector containing its similarity scores according to artwork, temporal distance, geographical distance and textual similarity. Each cast is then ranked according to its cosine similarity to the target cast.

Some of the data processing necessary to compute similarity was performed in

the Java program itself. The similarity according to time, location and artwork was performed in the Java program. For the Natural Language Processing (NLP) element of our system the Python programming language was used. Research into support for these techniques found that a number of suitable modules are available [41, 42, 43, 44, 45] which provide support for NLP, WSD, machine learning and document classification. The specific approaches taken to different aspects of the system are discussed below.

When discussing the implementation of algorithms the term "target cast" is used to describe the artcast upon which recommendations are based.

5.2.1 Time

For a target cast, the similarity according to time is computed using the Java 8 time and date API. The temporal distance between the target cast is given in days, and is converted to a similarity measure using equation 4.4.

5.2.2 Location

As discussed in the previous chapter, the haversine formula for computing distances between two points represented as longitude and latitude is used. The distance between a target artcast and all other casts is computed using this formula, and this is then converted into a similarity value using equation 4.4.

5.2.3 Artwork

The presence of a particular artcast is returned by a simple binary measure, 1 if the artwork of the target cast is present in other casts, 0 if not.

5.3 Serendipity

The Artcasting corpus provides a unique opportunity to implement a method of providing serendipitous results. If textual similarity is considered the principal measure of similarity, serendipitous artcasts can be returned according to their temporal and geographical distance and the artwork of an artcast. By retaining distance measures for temporal and geographical similarity, more distant artcasts are favoured. Similarly, limiting recommendations to artcasts which do not contain the artcast in the target cast In this way, artcasts which express similar sentiments but are sent to distant locations, and far forward or backward in time can be recommended to users. This assumes that location, date and artwork are good measures of serendipity, but in this particular context this seems a reasonable assumption to make.

5.4 Statistical Approaches to NLP

5.4.1 TF-IDF

As TF-IDF represents a standard NLP technique used by a number of applications, it was chosen for implementation in this system. A number of steps are taken in this system's approach to TF-IDF. First, we tokenise, stem and remove stopwords from our documents using tools provided by NLTK [41]. TF-IDF weights are then computed for all remaining words in the corpus, and documents are represented as sparse vectors of these values. For this the scikit-learn [43] Python library was used. The cosine similarity between the target document and all documents in the corpus was computed. This gives a value of 1 for a target artcast, and values in the range $[0, 1]$ for all other documents in our corpus.

Although it is possible to use training data for an implementation of TF-IDF, it was decided to use the Artcasting corpus in its entirety here. This allows for greater item-space coverage in a final implementation, as well as ensuring the algorithm is aware of all words in our corpus. In a larger corpus the use of training data may be more appropriate, but given the nature of our corpus it was important to extract a maximum of data from documents.

5.4.2 LDA

For an implementation of LDA, the gensim [44] library for Python was used. This library allows us to choose the number of topics the model identifies in a corpus. In our informal development testing phase, the model was asked to identify 2, 5, 10, 15 and 30 latent topics in individual tests in order to evaluate which number of topics returns topic which best represent the corpus.

At first the LDA model was trained on the small development and testing data set but as this set does not contain the complete set of topics available in the Artcasting corpus the model struggled to successfully identify the topics present in the entire corpus. Subsequently, the model was trained on the first 80 casts, and then the entire corpus. Each time the model was able to identify the topics of individual documents, but struggled to identify the topics in the corpus as a whole. As a result the implementation did not make classifications or find similarities between documents that make logical sense to a human observer. For example, when trained to find 15 latent topics the whole corpus the implementation assigns high similarity to the sentence pairs:

"Today is all about the music" and *"Reflections on the sea"*

And when trained to find 30 topics, returns:

"The Pop Art show, Roayal Acadamey was the first time I saw Lichenstein (and others) work up-close. Huge impression" and *"a warm january"*

as similar results. As many of these 'similarities' did not seem reasonable, it was decided not to continue with the implementation of LDA. The use of external training data may have improved the performance of this algorithm but this introduces irrelevant topics and may inappropriately weight the topics found in the Artcasting corpus. Although LDA is a suitable technique for large corpora, it is not appropriate for the relatively small corpus of short documents present in this situation.

5.5 Approach to Explicit Semantic Analysis

5.5.1 Word Sense Disambiguation

An effective technique for performing Word Sense Disambiguation (WSD) is essential for Explicit Semantic Analysis (ESA). Implementations of WSD algorithms were use from the pywsd [42] Python library.

There are a number of characteristics of the original Lesk algorithm [33] which make it useful for our recommender system. First of all, it is computationally cheap, processing user stories in an acceptable amount of time. However, its accuracy falls short of requirements. In the absence of definitions which are large or happen to contain specific words, it struggles to accurately disambiguate word senses. After some brief experimentation on a number of sentences in the devtest data set using the Lesk algorithm implemented in the Python Natural Language Toolkit module (NLTK) [41] it became apparent that the 50-70 % accuracy reported in some of Lesk's original experiments could not be replicated here. In the development and testing data set, an accuracy closer to 45% was reported. Although it could be argued that an appropriate word sense was found in some cases, another approach was sought. The adapted Lesk algorithm proposed by Banerjee and Pedersen [35] provides an alternative. When the same tests were run, an accuracy of between 60% and 70% was returned. Where WSD is performed, we use the adapted Lesk algorithm.

5.5.2 Similarity Techniques

The Python NLTK module [41] provides excellent support for a number of different similarity measures and a number of other modules are available for NLP using the Python programming language [42, 45]. This support made the Python programming language an obvious choice for our NLP techniques.

Previous studies have examined the suitability of a number of Wordnet techniques for computing similarity between word senses [32, 46, 47]. The Jiang-

Conrath measure is shown to perform well when compared to gold standard user-tagged data and was chosen for implementation. As it is an Information Content based technique, a technique which uses path similarity to compute similarity between word senses was chosen for comparison. The Leacock-Chodorow measure also performs well in the studies mentioned, and has also been applied in another similar project [4]. As a result, it was decided to implement these two measures for testing in our system. In this way the suitability of an IC based technique is compared against that of a path similarity based technique. In all implementations of Wordnet based techniques, Wordnet 3.0 was used.

5.5.3 The Brown Corpus

As the Jiang-Conrath technique depends on the Information Content (IC) of word senses, the Brown corpus [48] is used as an IC. This corpus contains probability tagged data from over 500 English language texts and numbers around 1,000,000 words. Although it is possible to create our own IC from the Artcasting corpus, there is no guarantee that this contains the IC of Lowest Common Subsumers necessary for the Jiang-Conrath algorithm. As the Brown corpus represents a standard source of Information Contents for word senses, it is used in this implementation.

5.5.4 Implementation of Similarity Techniques

Both the Jiang-Conrath and Leacock-Chodorow similarity measures were implemented in the same manner. For a document pair:

$$D_1 = (v_1, ..., v_n), D_2 = (w_1, ..., w_m) \tag{5.1}$$

the word sense of each word are disambiguated using the adapted Lesk algorithm [35] in order to obtain a representation of each document as their word senses, or synsets. This step gives:

$$D_1 = (L(v_1), ..., L(v_n)), D_2 = (L(w_1), ..., L(w_m)) \tag{5.2}$$

where $L(w_i)$ is the synset returned by performing the adapted Lesk algorithm on word w_i, using its the document D_i as context. For the document pair the similarity is then calculated as:

$$sim_D(D_1, D_2) = \frac{1}{mn} \sum_{i=0}^{n} \sum_{j=0}^{m} sim_s(L(v_i), L(w_j)) \tag{5.3}$$

for $v_i \in D_1$ and $w_i \in D_2$, and $sim_s(a, b)$ the similarity between synsets a and b.

This is performed for each pairing between the target document and documents in the corpus (including the target document). Once all similarities have been computed they are normalised to the range $[0, 1]$.

5.5.5 Semantic Similarity and Corpus Statistics

The final algorithm whose implementation is discussed is the one proposed by Li et al [31]. An implementation of this algorithm is available online [45]. For documents pairs, D_1, D_2 we are able to calculate similarity simply as $sim_{li}(D_1, D_2)$. Again, once these similarities have been computed between a target document and all corpus documents they are normalised in the range $[0, 1]$.

5.5.6 Using Document Similarity

As each of the NLP techniques used was implemented in the Python programming language, an effective method of transferring these similarity scores for use in our main Java program had to be implemented. By executing terminal commands from the Java program document similarity data was generated and read into the Java program.

5.6 Feature Vectors and Similarity

Each component of our system assigns a value, $v \in [0, 1]$, to each artcast in the corpus. These are then passed to a main "engine" module, which arranges these scores into feature vectors. Each feature vector describes the similarities of each cast's features to those of a target cast. Once artcasts have been scored according to one or more of artwork, geographical destination, temporal destination and textual similarity we create feature vectors for artcasts.

After running experiments with the different similarity measures discussed in the design chapter it was found all three techniques gave similar results when ranking artcasts according to their similarity. From the three techniques the cosine similarity was chosen. When recommending artcasts, the cosine similarity between the feature vector of our target cast (an all-ones feature vector) and that of each other artcast in the corpus is computed. The casts are then ranked according to their cosine similarity values.

Chapter 6

Evaluation and Findings

6.1 Evaluation Goals

In this chapter the performance of different aspects of the system are evaluated according to the methodology discussed in chapter 3. First, measures of serendipity are discussed and suggestions for their improvement are made. Next a brief discussion of time efficiency allows general suggestions for a deployment of this system to be made. A statistical analysis of the item space then helps evaluate the suitability of NLP algorithms for the Artcasting corpus. A discussion of the data gathered from user studies allows us to evaluate the suitability of algorithms and our entire system for deployment in the real world. Finally concrete findings are discussed.

6.2 Serendipity

Using the serendipity measures describe in our evaluation gives mixed results. As serendipity can be based on temporal distance, geographical distance, and artwork, each of these measures is discussed in turn.

Temporal distance is perhaps the least effective of the three criteria on which to base serendipity. As the majority of artcasts are sent to temporal locations close to the date on which they were cast, the temporal distances between them are small. Just a few artcasts are sent to vastly different temporal locations (extremes of the years 1602 and 2099), and it is these artcasts which are strongly favoured by an algorithm using temporal distance as a measure of similarity. By providing serendipitous results based on temporal distance, the item space coverage is reduced, as the few artcasts sent to distant times are consistently ranked the highest.

Geographical distance performs somewhat better as a measure of similarity, although suffers from a similar issue to that of temporal distance. The majority of artcasts are sent to locations around the UK, with a reasonable number sent further afield. Again, these more distant casts are favoured by an algorithm using geographical distance as a measure of serendipity when the target cast has been sent to the UK or Europe. Conversely, when artcasts sent far from the UK are chosen as the target cast, a large number of casts can be recommended. The geographical measure of serendipity is more successful than time-based serendipity as artcasts are sent to more diverse geographical locations than temporal locations.

Providing serendipitous results according to artworks is perhaps the most effective measure. As it is a simple binary measure, the recommendations provided exclude any artwork of the target cast. Although this may affect item space coverage and accuracy of items returned, if a user wishes to exclude the artwork of a target cast, this can be done successfully. Similarly, if users wish to view what users have said about a chosen artwork, this is performed successfully in the system.

In light of these findings, a refined method for filtering serendipitous results can be proposed. Firstly, allowing users to select the criteria by which artcasts are recommended allows users to explore the item space as they wish. Secondly, as textual similarity is the principal method for computing similarity between artcasts a system can be used to ensure serendipitous results are accurate. Only when the textual similarity of two artcasts exceeds a certain threshold is it appropriate to apply these serendipity measures. This ensures that serendipitous results are both surprising and accurate.

6.3 Time Efficiency

It is useful to measure the time efficiency of our NLP algorithms in order to make recommendations for future real-world implementations. The fastest algorithm to execute is TF-IDF, taking around 20 seconds to compute similarity between a target document and the corpus. The algorithms based on semantic similarity all take significantly longer. The Leacock-Chodorow algorithm performs next best, taking on average around 1 minute 30 seconds to compute similarity. The Jiang-Conrath algorithm took around 2 minutes to execute on the entire corpus, and the Li et al. algorithm took the longest, at around 4 minutes. All semantic similarity measures are dependent on the number of synsets returned by a particular document, and execute significantly faster for documents represented by a smaller number of synsets. However, this time is too long for practical deployment. Solutions to the issue of performance include the use of high performance data types and caching of models, synsets and cast weights in order to obtain faster performance in a deployment in the wild.

6.4 Document Length

One requirement for algorithms which compute document similarity is that comparable measures of similarity for both the long and short documents in our corpus are returned. This can be thought of as an approach to item-space coverage. As many documents have short length, in order to achieve high item-space coverage an algorithm must be able to generate similarity scores for these documents which are reasonably close to those generated for longer documents. To evaluate the performance of each of the four algorithms chosen for implementation (TF-IDF, Leacock-Chodorow, Jiang-Conrath and Li et al.), the average similarity score of each document when compared to all other documents in the corpus was computed for each algorithm. This average score is then plotted against the number of words contained in each document and correlations are examined.

When comparing the average document similarity to document length, it is expected that the average similarity increases somewhat with document length, as a longer document contains richer information from which an algorithm may draw data. Indeed, this correlation indicates that algorithms are returning meaningful data on document similarities. However, when examining the relationship between document similarity and document length we seek an algorithm that returns a comparable average similarity for long and short documents. This will appear as a shallow upwards curve in the correlation of the data points (indicating similarity doesn't increase much for longer documents), or large clusters of data points (indicating a number of documents are assigned similar similarity scores).

The first algorithm considered is TF-IDF (see figure 6.1). There is a weak correlation between document length and average document similarity. This shows that average similarities broadly increase with document length, although the specific behaviour is not predictable. It is also noted that the majority of documents are assigned a relatively low average similarity score. There are a large number of outliers, both with low document length and high similarity, and with high word count, but little similarity. By analysing the outliers the behaviour of this algorithm on our corpus can be better understood. The outlying texts chosen are assigned the highest similarity by TF-IDF:

1. *"beach: because he looks like he is chilling on the beach"*
2. *"maple syrup: she looks like she would enjoy some sugar, and this is a good place for that."*

and have document lengths of 11 and 18 words respectively. Both of these documents contain a high number of words which have high TF-IDF weightings. For example both the words *"beach"* and *"chilling"* occur just three times in the entire corpus, giving them high TF-IDF weights. The phrase *"looks like"*

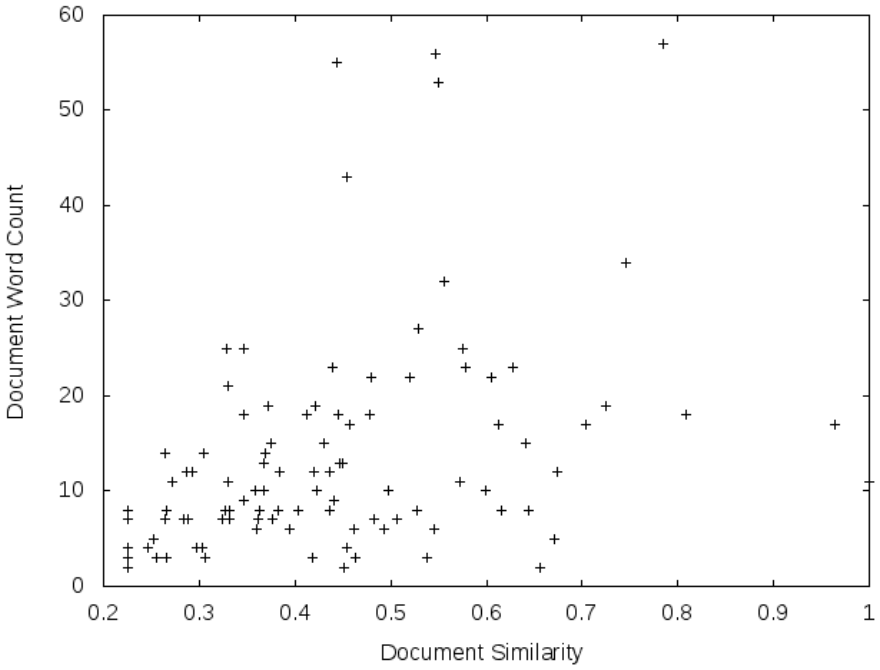

Figure 6.1: A plot of average TF-IDF document similarities against the length of documents in the Artcasting corpus

is relatively common, with the stemmed word *"look"* appearing 10 times and the word *"like"* appearing 12 times in the corpus. Although this is common for our corpus, these words would appear much more frequently in a larger corpus. Thus our corpus assigns them an artificially high weight. Similarly, in the 2^{nd} example, the words *"maple"*, *"syrup"* and *"sugar"* occur only once in the entire corpus. These terms are assigned a high weight and so the document is assigned a high average similarity.

The issues of both short document length and small corpus size contribute to the poor performance of TF-IDF. These qualities of the corpus cause relatively common words such as *"look"* and *"like"* to be attributed a high TF-IDF weight as they occur infrequently in our corpus. A corpus of longer documents would be more likely to contain common words such as these, increasing the document frequency and thus lowering the TF-IDF weight. The use of training data may solve this problem, but as performance on the corpus is already affected by the corpus size, separating training data from the data set prevents all words being assigned TF-IDF weights.

When average document similarity is plotted against document length for the

Jiang-Conrath algorithm (see figure 6.2), there is almost no correlation. Notably, there is a minimum threshold of around 0.18 and this is most likely the IC for a common Lowest Common Subsumer that occurs high up in the Wordnet ontology. Although comparable similarities are returned for documents of variable length, the lack of correlation and wide spread of the data indicates that this algorithm may not be providing reliable results. In order to better understand the situation, the document with highest length in our corpus is examined. This document has an average similarity of around 0.3 and is 57 words long.

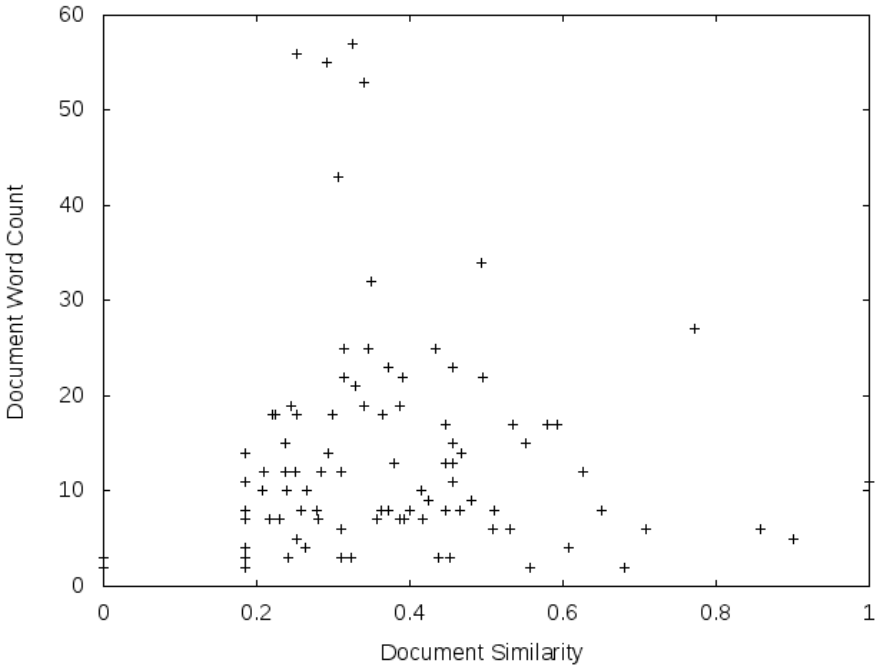

Figure 6.2: A plot of average Jiang-Conrath document similarities against the length of documents in the Artcasting corpus

"da Vinci hometown : i would like da Vinci to see how art is in the 21st century because I think he would love to get into photography. this reminds me of his drawing of the man in the circle. I love his cheeky face, like he's saying yes it's a classical reference but it's me as well."

There are clearly issues which influence the behaviour of the Jiang-Conrath algorithm on our corpus. Firstly, the Brown corpus [48] heavily influences the algorithm's behaviour. As the brown corpus is so large it is possible that the

values of Information Content of many word senses have similar values to those of their Lowest Common Subsumers. Although the idea expressed in this text is specific, the individual word senses are not. In the above example the deepest synsets in the Wordnet ontology are *"hometown"* and *"drawing"*, each with a depth of 11. The maximum depth of the noun taxonomy is 18, so although these synsets are reasonably specific, it is conceivable that they are not that rare Brown corpus. Further research is required to ascertain the exact reasons for the poor results of the Jiang-Conrath algorithm when applied to this corpus.

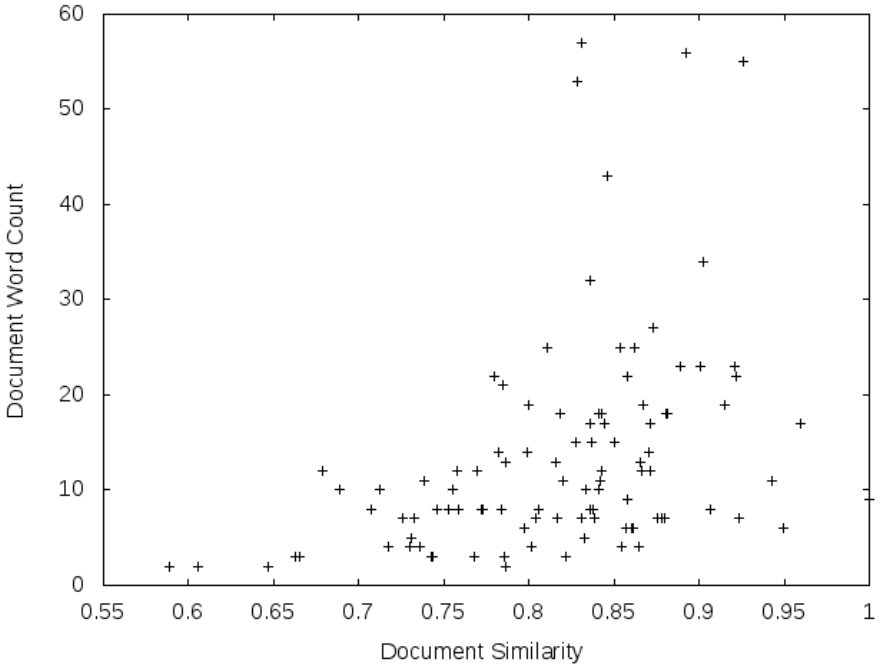

Figure 6.3: A plot of average Leacock-Chodorow document similarities against the length of documents in the Artcasting corpus

Next we consider the behaviour of the Leacock-Chodorow similarity measure when comparing the documents of our corpus (see figure 6.3). It is important to note that two outliers at 0 have been removed from the plot. These two entries correspond to cases where performing WSD on the documents returned no synsets. All other document similarities fall in the range $(0.55, 1]$ after normalisation between 0 and 1. Here we see that document similarity increases sharply with document length. The correlation is clearer than that of TF-IDF and Jiang-Conrath indicating the algorithm behaves reliably on the corpus and

that comparable similarity results are returned. However, the spread of the data indicates that documents are less likely to be assigned similar similarity scores than in our final algorithm.

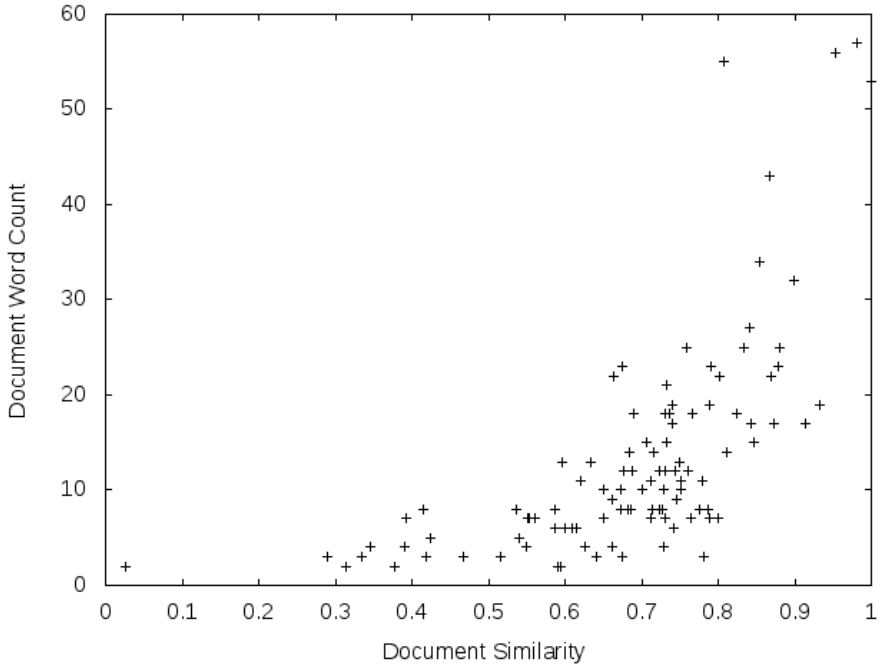

Figure 6.4: A plot of average Li et al. document similarities against the length of documents in the Artcasting corpus

The last algorithm considered here is that proposed by Li et al. (see figure 6.4. This algorithm exhibits the strongest correlation of data points between document length and average document similarity. Furthermore, a larger number of casts return comparable similarities scores. The cluster of points between 5 and 20 words indicates that the algorithm is able to assign comparable similarity scores to documents of differing length. The algorithm does however provide higher similarity scores to longer documents than Leacock-Chodorow. As these documents are not the norm in the corpus (there are only 6 documents longer than 30 words), it is felt that the strong clustering of points indicates that this measure is the most reliable measure in the face of varying document length.

6.5 User Studies

6.5.1 Respondents

User studies were first carried out at the Scottish National Gallery of Modern Art. Unfortunately an unusually quiet morning was chosen for the field surveys and only seven respondents were found at the gallery. As a result, fellow students at the University of Edinburgh were asked to fill in a surveys in order to reach a reasonable number. User studies were based on the responses of 14 individuals 5 females and 9 males. The majority of participants (6) were aged 18-24, 4 were aged 25-30, 1 was aged 31-40 and 3 were aged 41-50. Each respondent was asked for the average number of times they visited galleries per year in order to estimate their engagement with art. Of our 14 participants, 4 attended galleries 0-1 times a year, 5 attended 2-3 times, 4 attended 4-5 times and 1 attended 6-7 times per year. Respondents from the university and from the gallery visited galleries roughly the same amount per year.

6.5.2 Quantitative Data

Quantitative data was gathered by asking users about similar they perceived particular pairs of texts to be. This user-tagged data is taken as a gold standard as in previous studies [32]. There are a number of methods that can be used to to compute the effectiveness of the algorithms implemented for test. The root mean squared error is used to measure the accuracy of algorithms when compared with user-tagged data (see equation 3.1). This gives the accuracy of implemented algorithms when compared against user-tagged data. As each text is taken from percentiles of the data increasing according to average similarity, we expect some monotonically increasing relationship between algorithmic and user-tagged data. The Spearman correlation (see equation 3.4) is used to indicate the presence of this relationship. This gives the following results:

Algorithm Under Test	RMSE	Spearman Correlation
TF-IDF	0.494	0.702
Jiang-Conrath	0.466	0.157
Leacock-Chodorow	0.285	0.049
Li et al.	0.281	0.174

As can be seen, TF-IDF and Jiang-Conrath return the highest error when compared against user-tagged data. All algorithms except TF-IDF correlated weakly with the user-tagged data, but the Li et al. algorithm performs the best of the semantic measures. It is also observed that the Jiang-Conrath algorithm has the highest error (or lowest accuracy) of semantic similarity algorithms. The accuracy of the Leacock-Chodorow and Li et al. algorithms is similar, indicating that similarity measures which use path length and taxonomy depth may

be most appropriate for this context.

The high correlation coefficient for TF-IDF may be due in part to the large number of zero entries for document similarity present when the algorithm is executed on the texts for our surveys. By removing zero entries, a correlation coefficient of 0.674 is returned. This indicates that where TF-IDF is able to perform, it correlates well with user preferences. A similar value for accuracy is still given when these zero entries are removed. However this value is only based on 8 survey texts so it is difficult to draw firm conclusions.

It is noted that certain texts are assigned much higher similarity by users than by our algorithms. Some of the texts scored lowest by algorithms (those at the start of the survey) are assigned high similarity scores by users, particularly texts compared to the sentence:

"Down under: mid summer there -so jealous"

which reference Australia are assigned high similarity by users, but not by our algorithms. This is partly because the adapted Lesk algorithm used in our approach is unable to associated the phrase *"down under"* with Australia. In a second data analysis the 8 texts scored lowest by our algorithms are removed. This removes cases where word sense disambiguation was incorrectly performed on key words in a document. The following results are obtained:

Algorithm Under Test	RMSE	Spearman Correlation
TF-IDF	0.500	0.803
Jiang-Conrath	0.534	-0.012
Leacock-Chodorow	0.311	-0.139
Li et al.	0.229	0.437

The notable difference here is that the Li et al. algorithm exhibits the greatest monotonic relationship with user preferences. Although the correlation is not particularly strong it is certainly present.

From the quantitative data obtained on user interest it is noted that user interest in the similarity of a particular pair does not always correlate with users assigning that pair high similarity. Often interest in one of the two documents in the pair suffices for users to indicate an interest in the overall similarity. This suggests that personal experience plays a role in user interest in document similarity. This is discussed further in the following section.

This data analysis shows quite clearly that the Li et al. algorithm exhibits the greatest accuracy when compared with user-tagged data, and correlates the strongest with user preferences. As a result this is the textual similarity algorithm most suited for use in a recommender system for the Artcasting application.

6.5.3 Qualitative Data

Qualitative data was gathered from survey participants in order to assess the impact of a recommender system for the Artcasting application and to support findings from quantitative analyses. The data was gathered in two main ways. Firstly by asking if they have interest in the similarity between particular pairs of texts from the corpus and secondly by asking them to explain why they found a particular pair of casts interesting.

The first thing we note is that there are a large number of comments, representing a number of reasons for users finding similarities interesting and relevant. Users made comments that the suggested similarities would inspire them to read novels and visit new locations, as well as remind them of their past experiences and people who inspire them. Comments were left for a variety of reasons. For example, for the pair

"School : The text on the jacket reminds me of the effort I would put into scrawling my favourite bands' names all over books and pencil cases. You can see how much music is a visual part of someone's identity, especially at a young age and this was very important to me growing up" and *"Kamron : I sent this picture because I am in a music school, and because I love music"*

all 5 comments relate to the common words in the document pair. Similarly, many comments give the semantic themes of the texts as the source of their interest. This shows the chosen algorithms were appropriate for this context.

Other comments on the interest in the similarity of a document pair relate to personal experiences For example, for the document pair

"S'Algar Diving Centre : This brings back memories of watching water for hours as my brother learned to dive. Admittedly, Lichtenstein's influence was Giverny, and the calm water of a lily pond rather than the Mediterranean Sea. But the foil sections of this work make me think of waves and mermaids' tales flapping beneath the water's surface" and *"on the way to the little mermaid : walking to see the little mermaid"*

a number users gave the reasoning for finding interest in the pair as an interest in fantasy themes.

In general users who attended galleries more often were more likely to leave comments relating text similarities to personal experience, although both were about equally likely to express interest in textual similarities. This suggests that an information filtering system providing non-personalised feedback may increase user engagement for those less likely to attend a gallery. However, the presence of personal explanations indicates that personalised recommendations may be useful for a recommender system for Artcasting.

6.5.4 Issues

After giving a sample questionnaire to a few colleagues, feedback was that it took too long to fill out. The 4th and 6th percentile texts were removed in an effort to reduce the time taken to fill out the questionnaire. In this way, the questionnaire length was reduced. The 4th and 6th percentiles were chosen for removal in order to allow us to study the behaviour of documents most and least likely to be calculated as similar while retaining some data for documents which have an average likelihood of being rated as similar by the chosen algorithms.

One issue was participants' understanding of what was being asked of them. Effort was made to explain the survey clearly to participants, but participants were not always sure how to proceed. This represents shortcomings in the ability of the survey to effectively communicate its purpose. It is true that some documents from the Artcasting corpus can be confusing or appear to make little sense when viewed out of context. Despite these shortcomings the surveys were completed in a satisfactory manner, and trends in the data indicate that respondents eventually understood what was asked of them, even if it was not immediately clear to all.

Selecting appropriate documents for user comparison is another issue in the evaluation of this project. As TF-IDF approaches document similarity in a different way to the other algorithms under test, evaluating this measure in the same way as the the other techniques in this project is not straightforward. Indeed, as the majority of the documents chosen for evaluation do not contain common words, we struggle to properly evaluate the TF-IDF algorithm against user-tagged data. However, the evaluation of item-space coverage indicates other algorithms are more appropriate for this particular situation.

6.6 Discussion

It was hypothesised that techniques for semantic analysis would perform better than others over the short documents in the Artcasting corpus. The impact of implementing these techniques on user engagement was also sought. The user survey gave quantitative feedback on the appropriateness of these techniques as well as allowing the impact of these techniques to be gauged. An in depth quantitative analysis of the item-space coverage of these techniques allowed for further insight into their behaviour and appropriateness for an information filtering system for Artcasting. Novel methods of providing serendipitous results were introduced and evaluated.

Preliminary investigations showed that an implementations of Latent Dirichlet Allocation on the Artcasting corpus was not appropriate to its size and the length of documents contained within it. By evaluating TF-IDF against algorithms which use the Wordnet database to compute similarity between doc-

ument pairs we ascertain that although TF-IDF provides some good results, its ability to provide a large number of meaningful results is limited by the distribution of words throughout the Artcasting corpus. Techniques which use semantic similarity techniques do generally perform better than those that do not, but the results are not unequivocal.

The edge counting methods implemented here perform significantly better than the node-based Jiang-Conrath algorithm. It is shown that edge counting methods give more stable results over varying document length and a lower error when compared to gold standard user data. However, when compared with TF-IDF it is shown that text weighting methods may correlate with user preferences. The results presented to this effect are not particularly strong and further research is required to understand the full effect of term weighting on our corpus.

The best performing algorithm on the Artcasting corpus was that proposed by Li et al., and it is strongly suggested to use this algorithm in a recommender system for Artcasting. This algorithm has the lowest rate of error when compared with other algorithms implemented and the strongest correlation with user preferences.

The implementation of techniques for providing serendipitous add an interesting dimension to recommendations from the Artcasting corpus. They are relatively simple to implement and add to the playful and creative nature of the Artcasting application. Although some care must be taken to ensure the implementation of these techniques does not affect the accuracy of results returned, these techniques may be implemented successfully in a recommender system for Artcasting.

Finally, qualitative feedback from users shows that an information filtering for Artcasting does provide interesting comparisons between artcasts. Users showed interest in textual similarity in a range of ways. Many users were interested in items which related to their own experiences. The presence of personal opinion in feedback on text similarity suggests that implementing a true recommender system, with personalised results would be beneficial to this project. This is not to underestimate the impact of the information filtering system presented in this project. The feedback gained from users studies indicates that appropriate trade-offs in development and evaluation were made and that the system proposed here does well to engage users.

Chapter 7

Conclusions

The research in this dissertation investigated the implementation and impact of similarity techniques for use in an information filtering system for Artcasting. A system was built to incorporate a range of similarity techniques, some seen in other works and others unique to the Artcasting corpus. The project shows that an information filtering system based on textual similarity of artcasts for the Artcasting application can successfully be implemented on the user-generated corpus of artcasts. User surveys aided in the evaluation of an appropriate textual similarity algorithm. The hypothesis that semantic similarity techniques aligned best with user-judged similarity on our small corpus was proven. Specifically the algorithm proposed by Li et al. [31] provides the best measure of document similarity for our corpus. This algorithm uses path-based semantic similarity and word order to compute similarity between documents. The user study also provided valuable qualitative feedback on the impact of implementing a recommender system for Artcasting. The user study conducted in this project indicates that the filtering of results increases user engagement with the Artcasting application, justifying the choices made in this project.

The user survey conducted in this project indicated that personalised results may provide greater impact in the Artcasting recommender system. Similar projects focus on the personalisation of results [4, 5], however other research has put forth the idea that systems which present recommendations to users should guide users to discover new works rather than display results strictly based on information users enter into a system [1]. Future research into the trade-off between the freedom to explore all items in a collection and the personalisation of results would help systems provide users with appropriate interfaces for the discovery of new items of cultural heritage.

Using semantic similarity techniques proved successful in this project, however some cultural references proved difficult to analyse. Further research into techniques seen in previous studies [1] using online databases to compute similarity

may prove a successful approach to future works which involve providing recommendations to users.

This project provides a firm basis for expansion to the functionality of the Artcasting application. The system proposed here provides a method of computing similarity between documents in the Artcasting corpus, as well as methods of providing serendipitous results to users. As the system built here functions as a back end to a recommender system, developing appropriate user interfaces would be the next step in the development of this project.

Bibliography

[1] K. Grieser, T. Baldwin, F. Bohnert, and L. Sonenberg, "Using ontological and document similarity to estimate museum exhibit relatedness," *J. Comput. Cult. Herit.*, vol. 3, no. 3, pp. 10:1–10:20, Feb. 2011. [Online]. Available: http://doi.acm.org/10.1145/1921614.1921617

[2] A. L. Cushing, "Theorizing digital cultural heritage: A critical discourse," *The American Archivist*, vol. 72, no. 1, pp. 250–252, 2009. [Online]. Available: http://www.jstor.org/stable/40294611

[3] "Moma iphone application," https://www.moma.org/explore/mobile/iphoneapp, accessed: 13-08-2016.

[4] G. Semeraro, P. Lops, M. De Gemmis, C. Musto, and F. Narducci, "A folksonomy-based recommender system for personalized access to digital artworks," *J. Comput. Cult. Herit.*, vol. 5, no. 3, pp. 11:1–11:22, Oct. 2012. [Online]. Available: http://doi.acm.org/10.1145/2362402.2362405

[5] F. Bohnert and I. Zukerman, *Non-intrusive Personalisation of the Museum Experience*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 197–209. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-02247-0_20

[6] T. Moussouri and G. Roussos, "Conducting visitor studies using smartphone-based location sensing," *J. Comput. Cult. Herit.*, vol. 8, no. 3, pp. 12:1–12:16, Mar. 2015. [Online]. Available: http://doi.acm.org/10.1145/2677083

[7] A. Wingard, "Informatics research proposal: Finding novel ways of interacting with user sentiments and art with a recommender system for artcasting," April 2016.

[8] "Artcasting project," https://www.artcastingproject.net/, accessed: 06-08-2016.

[9] "Artist rooms," http://www.artistrooms.org/, accessed: 06-08-2016.

[10] M. Sheller and J. Urry, "The new mobilities paradigm," *Environment and Planning A*, vol. 38, no. 2, pp. 207–226, 2006. [Online]. Available: http://EconPapers.repec.org/RePEc:pio:envira:v:38:y:2006:i:2:p:207-226

[11] "Google art project," https://www.google.com/culturalinstitute/beta/, accessed: 06-08-2016.

[12] "History pin," https://www.historypin.org/en/, accessed: 06-08-2016.

[13] "Magic tate ball," http://www.tate.org.uk/context-comment/apps/magic-tate-ball, accessed: 06-08-2016.

[14] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, *Recommender Systems Handbook*, 1st ed. New York, NY, USA: Springer-Verlag New York, Inc., 2010.

[15] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, Jan. 2003. [Online]. Available: http://dx.doi.org/10.1109/MIC.2003.1167344

[16] "last.fm," http://www.last.fm/, accessed: 13/08/2016.

[17] "Rotten tomatoes," https://www.rottentomatoes.com/, accessed: 13/08/2016.

[18] "Pandora radio," http://www.pandora.com/, accessed: 13/08/2016.

[19] "Imdb," http://www.imdb.com/, accessed: 13/08/2016.

[20] Y. Wang, N. Stash, L. Aroyo, L. Hollink, and G. Schreiber, "Using semantic relations for content-based recommender systems in cultural heritage," in *Proceedings of the 2009 International Conference on Ontology Patterns - Volume 516*, ser. WOP'09. Aachen, Germany, Germany: CEUR-WS.org, 2009, pp. 16–28. [Online]. Available: http://dl.acm.org/citation.cfm?id=2889761.2889763

[21] G. Benelli, A. Bianchi, P. Marti, E. Not, and D. Sennati, "Hips: hyper-interaction within physical space," in *Multimedia Computing and Systems, 1999. IEEE International Conference on*, vol. 2, Jul 1999, pp. 1075–1078 vol.2.

[22] G. A. Miller, "Wordnet: A lexical database for english," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995. [Online]. Available: http://doi.acm.org/10.1145/219717.219748

[23] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, ser. IJCAI'07. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2007, pp. 1606–1611. [Online]. Available: http://dl.acm.org/citation.cfm?id=1625275.1625535

[24] L. M. Eranti, V., "The social significance of the facebook like button," *First Monday*, vol. 20, no. 6, 2015.

[25] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003. [Online]. Available: http://dl.acm.org/citation.cfm?id=944919.944937

[26] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, no. 6, pp. 391–407, 1990.

[27] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ser. ACL '04. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004. [Online]. Available: http://dx.doi.org/10.3115/1218955.1218990

[28] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *Processing*, pp. 1–6, 2009. [Online]. Available: http://www.stanford.edu/~alecmgo/papers/TwitterDistantSupervision09.pdf

[29] C. Salton, G. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, Aug. 1988. [Online]. Available: http://dx.doi.org/10.1016/0306-4573(88)90021-0

[30] W. Hua, Z. Wang, H. Wang, K. Zheng, and X. Zhou, "Short text understanding through lexical-semantic analysis," in *2015 IEEE 31st International Conference on Data Engineering*, April 2015, pp. 495–506.

[31] Y. Li, D. McLean, Z. A. Bandar, J. D. O'Shea, and K. Crockett, "Sentence similarity based on semantic nets and corpus statistics," *IEEE Trans. on Knowl. and Data Eng.*, vol. 18, no. 8, pp. 1138–1150, Aug. 2006. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2006.130

[32] A. Budanitsky and G. Hirst, "Evaluating wordnet-based measures of lexical semantic relatedness," *Comput. Linguist.*, vol. 32, no. 1, pp. 13–47, Mar. 2006. [Online]. Available: http://dx.doi.org/10.1162/coli.2006.32.1.13

[33] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone," in *Proceedings of the 5th Annual International Conference on Systems Documentation*, ser. SIGDOC '86. New York, NY, USA: ACM, 1986, pp. 24–26. [Online]. Available: http://doi.acm.org/10.1145/318723.318728

[34] A. Clark, C. Fox, and S. Lappin, *The Handbook of Computational Linguistics and Natural Language Processing*. Wiley-Blackwell, 2010.

[35] S. Banerjee and T. Pedersen, "An adapted lesk algorithm for word sense disambiguation using wordnet," in *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, ser. CICLing '02. London, UK, UK: Springer-Verlag, 2002, pp. 136–145. [Online]. Available: http://dl.acm.org/citation.cfm?id=647344.724142

[36] C. Leacock and M. Chodorow, "Combining Local Context and WordNet Similarity for Word Sense Identification," in *WordNet: An electronic lexical database.*, C. Fellbaum, Ed. MIT Press, 1998, ch. 13, pp. 265–283.

[37] P. Resnik, "Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language," *CoRR*, vol. abs/1105.5444, 2011. [Online]. Available: http://arxiv.org/abs/1105.5444

[38] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," *CoRR*, vol. cmp-lg/9709008, 1997. [Online]. Available: http://arxiv.org/abs/cmp-lg/9709008

[39] D. Lin, "An information-theoretic definition of similarity," in *Proceedings of the Fifteenth International Conference on Machine Learning*, ser. ICML '98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 296–304. [Online]. Available: http://dl.acm.org/citation.cfm?id=645527.657297

[40] "Beautiful soup html parser," https://www.crummy.com/software/BeautifulSoup/, accessed: 09-08-2016.

[41] E. Loper and S. Bird, "Nltk: The natural language toolkit," in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ser. ETMTNLP '02. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 63–70. [Online]. Available: http://dx.doi.org/10.3115/1118108.1118117

[42] L. Tan, "Pywsd: Python implementations of word sense disambiguation (wsd) technologies [software]," https://github.com/alvations/pywsd.

[43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[44] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50, http://is.muni.cz/publication/884893/en.

[45] S. Patel, "Implementation of the algorithm for sentence similarity proposed by li et al. [31]," http://sujitpal.blogspot.co.uk/2014/12/semantic-similarity-for-short-sentences.html, accessed: 10-08-2016.

[46] G. Varelas, E. Voutsakis, P. Raftopoulou, E. G. Petrakis, and E. E. Milios, "Semantic similarity methods in wordnet and their application to information retrieval on the web," in *Proceedings of the 7th Annual ACM International Workshop on Web Information and Data Management*, ser.

WIDM '05. New York, NY, USA: ACM, 2005, pp. 10–16. [Online]. Available: http://doi.acm.org/10.1145/1097047.1097051

[47] H. Li, Y. Tian, B. Ye, and Q. Cai, "Comparison of current semantic similarity methods in wordnet," in *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, vol. 4, Oct 2010, pp. V4–408–V4–411.

[48] W. N. Francis and H. Kucera, "Brown corpus manual," Department of Linguistics, Brown University, Providence, Rhode Island, US, Tech. Rep., 1979. [Online]. Available: http://icame.uib.no/brown/bcm.html

Chapter 8

Appendices

Research Study: Exploring Interesting connections through sentence similarity

Hello and thank you for taking the time to help me with my research. I am gathering data for my MSc dissertation in Computer Science. For this project I am working with a mobile application, called Artcasting which examines how visitors experience art exhibitions (more information available here: https://www.artcastingproject.net/). The app was piloted here at the Gallery earlier this year and my project is building on the pilot to explore ways that people might connect with 'artcasts' other users have sent.

The artcasting application invited users to 'cast' an artwork they saw in the gallery to a particular time and place, with a short message (like a postcard). My research focuses on processing the messages people write and computing the similarity between these messages to encourage engagement with the application.

Below are a number of artcasting descriptions written by users of the app. There are 8 principal artcasts, and below each of these there are 4 further artcasts. Please read each principal message and score its similarity to each of the 4 messages underneath it 1-10 (1 being least similar, 10 being most similar). How you think about similarity is up to you: it could be the specific words used in a text, general sense of the text, may involve cultural references, or may relate to your own experiences, views or beliefs.

Once you have given a number for similarity please indicate whether you feel the link would encourage you to further explore other information (pictures, artworks, dates, places) associated with either text. If you feel the link may not be instantly obvious from context please explain it under why/why not.

This project has received ethical approval from the School of Informatics, University of Edinburgh. You can stop participating at any time, and the information you share will be totally anonymous. If you have any questions, you can contact me at alex.wingard92@gmail.com

First please enter some information about yourself (leave blank if prefer not to say):

Age Range:	<18	18-24	25-30	31-40	41-50	50+
Sex:	male	female	other			
On average, how many times a year do you visit art galleries?:	0-1	2-3	4-5	5-6	6-7	7+

Text A:

Down under mid summer there -so jealous

1) Adelaide hello Adelaide

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

2) I almost went like a missed opportunity, almost clear.

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

3) jazz hands Jazz hands on Broadway, naturally

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

4) Maryam because it looks something from the rainforest in Australia

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

Text B:

Houston launched many small rockets from here with the kids, lost a few too!
1) westmister explosions Similarity to text B (1-10): Does the link between the sentences interest you? (Y/N): Why?
2) Holidays good memories Similarity to text B (1-10): Does the link between the sentences interest you? (Y/N): Why?
3) S'Algar Diving Centre This brings back memories of watching water for hours as my brother learned to dive. Admittedly, Lichtenstein's influence was Giverny, and the calm water of a lily pond rather than the Mediterranean Sea. But the foil sections of this work make me think of waves and mermaids' tales flapping beneath the water's surface. Similarity to text B (1-10): Does the link between the sentences interest you? (Y/N): Why?
4) Robbie whelton I like the picture and he really likes playing the piano and he is having a baby and I thought he could show it to his little girl when it is born Similarity to text B (1-10): Does the link between the sentences interest you? (Y/N): Why?

Text C

Amber warning of wind I am windswept and on my way

1) School of Design about to head off to an undergraduate exam board...

Similarity to text C (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

2) Mick and Gwen Davies because Granny and Grandpa are in the car

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

3) Perth rainy places need dry art

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

4) aurora borealis I hope to see it again...

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

Text D

<u>Miss Havisham</u> for some reason I ended up inside the most extraordinary house in Kibworth that was owned by a Miss Havisham who toured me through the memorabilia of her long lost husband who had fought in the boer war. Out of time but not out of touch, the house was between life and death.
1) <u>Van Gogh museum</u> memory of her concert Similarity (1-10): Does the link between the sentences interest you? (Y/N): Why?
2) <u>New York</u> I just thought it fit with the sexual politics frequently brought up in the news today Similarity (1-10): Does the link between the sentences interest you? (Y/N): Why?
3) <u>Place of defiance</u> I believe that if I go to Vancouver, I will learn to become my own person and defy stereotypes like this photo Similarity (1-10): Does the link between the sentences interest you? (Y/N): Why?
4) <u>Buckingham palace</u> The queen needs to see some eyes Similarity (1-10): Does the link between the sentences interest you? (Y/N): Why?

Text E

S'Algar Diving Centre This brings back memories of watching water for hours as my brother learned to dive. Admittedly, Lichtenstein's influence was Giverny, and the calm water of a lily pond rather than the Mediterranean Sea. But the foil sections of this work make me think of waves and mermaids' tales flapping beneath the water's surface.

1) tattoo fixers wondering if he would 'fix' any of his tattoos?

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

2) soup can to be reunited with a soup can, way up high

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

3) beach reflections on the sea

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

4) on the way to the little mermaid walking to see the little mermaid.

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

Text F

School The text on the jacket reminds me of the effort I would put into scrawling my favourite bands' names all over books and pencil cases. You can see how much music is a visual part of someone's identity, especially at a young age and this was very important to me growing up.

1) Mount Fuji Japan is the best

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

2) music to lift off to Some jazz for Timothy Peake during blast off.

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

3) Arnie Sometimes you just need a strong guy.

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

4) Kamron I sent this picture because I am in a music school, and because I love music

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

Text G

da Vinci hometown i would like da Vinci to see how art is in the 21st century because I think he would love to get into photography. this reminds me of his drawing of the man in the circle. I love his cheeky face, like he's saying yes it's a classical reference but it's me as well.

1) self presentation a reminder to be self possessed & confident when giving my presentation today.

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

2) Edinburgh college of art it can join the nudes in the sculpture court, the casts are currently not on display so here are two nudes

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

3) Amsterdam as I love the photo because there is something simplistic and lovely about it, and I will be there

Similarity (1-10):

Does the link between the sentences interest you? (Y/N):

Why?

4) first times possibly the first time I saw Lichtenstein in person but a memorable trip to London

Similarity (1-10)

Does the link between the sentences interest you? (Y/N):

Why?

Text H

beach because he looks like he is chilling on the beach
1) cold and hot reflections Similarity (1-10) Does the link between the sentences interest you? (Y/N): Why?
2) central park Chilling out in the park Similarity (1-10) Does the link between the sentences interest you? (Y/N): Why?
3) aurora borealis I hope to see it again... Similarity (1-10) Does the link between the sentences interest you? (Y/N): Why?
4) gallery of art I wanted to communicate this picture with my colleagues in Greece Similarity (1-10) Does the link between the sentences interest you? (Y/N): Why?

Ethical Review Procedures: Level 1

Project Details & Self-assessment

This document is closely modelled on documents used in School of Philosophy, Psychology and Language Sciences provided by Ellen Bard and Cedric MacMartin.

This form is to be filled in and submitted at the same time as the project proposal or the funding application it applies to. The form should be submitted by the Principal Investigator, except in the following cases:

- Post-doctoral fellowships – the proposed postdoc mentor.
- UG, MSc, and PhD research projects – the supervisor.
- Visiting researcher – the staff hosting the visitor.

Please submit the completed form by email to: **infkm+ethics@inf.ed.ac.uk**

This address, with appropriate RT number once issued, should be used for all correspondence (including forms and attached documents). This is essential to ensure proper record keeping. No signature is required if the form is sent from a valid University email address.

Project Details

1 Type Of Project:

☐ Research grant proposal ☐ UG final year project ☑ MSc project

☐ Post-doctoral fellowship ☐ PhD project ☐ Research performed by visiting researcher

☐ Personal research ☐ Other: _____

2 Is there a sponsor/ funding body? YES / NO

3 Does the sponsor/funder require formal prior ethical review? YES / NO
If yes, by what date is a response required ?

4 Is any other institution and/or ethics committee involved? YES / NO
If YES, give details and indicate the status of the application at each other institution or ethics committee (i.e., submitted, approved, deferred, rejected):

5 Title of Project An Exploration of Item Relatedness in a User-Curated Corpus of Digital Cultural Heritage

6 Researchers' names, affiliations, emails
Include student/supervisor, post-doc/mentor, PI, or visitor/host. Dragan Gasevic dragan.gasevic@ed.ac.uk
Jen Ross jen.ross@ed.ac.uk

7 State which professional organisation guidelines you are using:

☑ School of Informatics research ethics code: http://www.inf.ed.ac.uk/research/ethics/

☐ Other ethics code as required by funding body or professional organization:
Title: _____ URL: _____

Self-assessment

Refer to Level 2 form for details on any of the following points.

1. **Protection of research participants' confidentiality**
   Are there any issues of CONFIDENTIALITY which are NOT ADEQUATELY HANDLED by normal tenets of academic confidentiality? YES / (NO)

   These include well-established sets of procedures that may be agreed more or less explicitly with collaborating individuals/organisations, for example, regarding:
   (a) Non-attribution of individual responses;
   (b) Individuals and organisations anonymised in publications and presentation;
   (c) Specific agreement with respondents regarding feedback to collaborators and publication.

2. **Data protection and consent**
   Are there any issues of DATA HANDLING AND CONSENT which are NOT ADEQUATELY DEALT WITH, and compliant with established procedures? YES / (NO)

   These include well-established sets of procedures, for example regarding:
   (a) Compliance with the University of Edinburgh's Data Protection procedures (see http://www.recordsmanagement.ed.ac.uk);
   (b) Respondents giving consent regarding the collection of personal data (via consent form).

3. **Significant potential for physical or psychological harm, discomfort or stress**
   Are there any risks of :
   (a) psychological harm or stress for the participants? YES / (NO)
   (b) physical harm or discomfort for the participants? YES / (NO)
   (c) any kind to the researcher? YES / (NO)

4. **Vulnerable participants**
   Are any of the participants in the research vulnerable, e.g., children, patients, disabled participants? YES / (NO)

5. **Moral issues and researcher/institutional conflicts of interest**
   Are there any SPECIAL MORAL ISSUES/CONFLICTS OF INTEREST? These include:
   (a) Conflict of interest: potential benefit to the researcher, friends or family of a particular research outcome which might compromise the researcher's objectivity or independence;
   (b) The need to keep the purposes of research concealed;
   (c) Use of participants who are unable to provide informed consent (e.g., children);
   (d) Situations where research findings would impinge negatively/differentially upon the interests of participants.

   YES / (NO)

6. **Bringing the University into disrepute**
   Is there any aspect of the proposed research which might bring the University into disrepute? For example, could any aspect of the research be considered controversial or prejudiced? YES / (NO)

7. **Use of animals**
   Does the research involve animals? YES / (NO)

8. **Developing countries**
   Does the research involve developing countries? YES / (NO)

21/1/16

9. Dual use

Is the research classified or does it have specific adversarial military applications? YES / (NO)

10. Terrorist or extremist groups

Does your research concern groups which may be construed as terrorist or extremist? YES / (NO)

Can you stop now?

You may want to assure yourself that your 'NO' answers are correct by checking the detailed form in the next section.

If all the YES / NO answers are NO, the self assessment has been conducted and confirms the ABSENCE OF REASONABLY FORESEEABLE ETHICAL RISKS. This form should be signed by the researchers and submitted. The researchers may retain a copy for their own records.

If any answer is YES, please complete the relevant section in the Level 2 form below.