Comparing the qualitative impact of different features and similarities on fictional text using SIMFIC

Sayantan Polley and Suhita Ghosh

Otto-von-Guericke University, Magdeburg
sayantan.polley@st.ovgu.de,
suhita.ghosh@st.ovgu.de,

Abstract. The goal of this paper is to present: SIMFIC, a text retrieval system for searching English fiction books, using selected 19th century books from 'Project Gutenberg' website. We have extracted features that address various qualitative aspects of literature like writing style, complexity, sentiment and others to create an index of the corpus. Users make a query by example (QBE), where a query is a whole book. We have divided the query book into small sections called 'chunks'. We measure the similarities of chunks of the selected query book with the chunks of all other books of the corpus, using various known text similarity measures. We experimented and devised ways to finally aggregate the similarities between chunks to create the final relevance rank list of books. We call the prototype based on our way of finding similarity as SIMFIC ('similarity in fiction'). There is a second variant of SIMFIC, which displays a short justification of the search results, based on feature selection. We evaluate SIMFIC by comparing with Apache Lucene (third system) in a user study. We have three systems on a corpus of about a thousand books. The user study is undertaken by a select cohort of English fiction experts (bachelor, master students and professors, at a university Department of English). In the user study, we deliberately kept a section on 'known popular books' that are taught to the students of English in their program. In other section, participants could select a book of their choice. The user study shows promising evidence that the SIMFIC prototype built on hand crafted features was an effective system to aid the information search needs on fiction book corpus.

Keywords: Text Retrieval, English Fiction, Gutenberg

1 Introduction

As we know a great book leaves us with many experiences, slightly exhausted at the end, more exhausted while searching a book of our interest. Borrowing books and leisure reading are popular activities and most of the books borrowed in the public libraries are fiction [13][6]. The provision of fiction literature in the public library seemed to get a negative attitude from the librarians until 19th century. Novels were accepted in the library for the sake of educational value, especially for the 'lower classes' in the late 19th century. The attitude mellowed down with the change of time and social values. Fiction literature is being read by 80% of Finns in 2010 [25] and at least once a year by 45.2% of Americans in 2012 [14] . In the Netherlands, the share of all borrowed fiction books was 82% in 2015 [5], 43% in United Kingdom in 2013 [29] and 68% in Finland in 2017 [13]. Until the advent of digital era, public libraries served as the primary source of getting access to fiction literature. A library user typically searches for a book based on its meta data like title or author [10] which often does not cater to users' need or the user is deprived of diverse results. Hence, a section of people who prefer to go to libraries rather than getting it over Internet, might prefer browsing bookshelves rather than surfing the library database [7][10]. Also, studies shows that the on-line library catalogues are preferable for known author or title searches, but not as effective in browsing books [1]. There have been efforts on content based book search. For example, *BookSampo Project* [9] implemented a semantic web portal enabling search beyond meta-data based on ontologies.

The aim of this paper is to explore and compare the qualitative features of fiction text that aids an user for searching similar books, given a query book. To support this idea we have extracted features and propose a similarity model based on existing text similarity measures. We have implemented a search prototype based on our model for 19th century fiction book corpus *Project Gutenberg* using content based features other than the usual keyword and meta data search. We evaluate and qualitatively compare our model (we call it SIMFIC) with a popular search engine (Apache Lucene) based on the classic vector space model. Non-meta

data content based features might help the user in getting helpful results which would cater to user needs and at the same would retrieve some books which would be perceived by the user as a pleasant surprise.

The paper is structured as follows: we start with a short summary of related literature in this domain in section two. Section three explain the overall concept of our task and thought process behind the choice of features. Section four describe the corpus, preprocessing and feature extraction. We describe the SIMFIC algorithm, similarity measure and feature selection in section five. Section six explain the user study setting with participant and faculty demographics. Section seven contain evaluation results and analysis of the user study. Final sections (eight and nine) discuss the limitations of our work, possible future research work and conclusion.

2 Related Work

Fiction literature has found its place in the public libraries and into our bookshelves. However, not much work has been done to cater to the information needs of the fiction readers using fiction search engines. Many studies have been performed on selecting fiction in physical collections [7], but relatively less on how readers access novels using fiction search engines [11] and evaluation for the same.

Mikkonen and Vakkari [10] found the most common method to search book in public library system by using author or title search, after performing a survey over Finnish population aged 15 to 79 years. This method was used by 57% of the respondents. Spiller [27] also found most of the readers (54%) select books by searching on metadata like author's name or title, compared to 46%, who searched novels by browsing in the library. 78% of the users found the books by using the combination of the two approaches. If users were not aware of the author while searching, 88% of the users selected books based on the text on the back cover, describing the book and the author. Apart from that, the users selected books after reading through the text passages from the book (33%) and title (22%) [27]. Spiller's study was replicated by Davidson and Cave [3] in New Zealand a decade later and also found users resorted to the basic metadata for selecting books when done through the online catalogue.

Oksanen and Vakkari [16] studied the search moves in an enriched public library catalogue. They conducted the study over 58 users who had vague idea about their user need. They found that advanced search, scanning book pages and exploring the enriched search results were the primary search tactics adopted by the users. Their investigations found that the efforts expended over examining results and inspecting book pages improved the chances of finding interesting novels instead of querying in browsing conditions. Though an advanced search might have helped an user in getting diverse results but a study done by Mikkonnen and Vakari [11] found that the users considered formulating queries complex other than for a known author or title. In a gaze tracking study, Pöntinen and Vakkari [20] studied how a book's metadata is being used by readers for a traditional and an enriched catalog when searching for fiction. They had recorded and analyzed the eye movements of 30 users while selection of fiction. They found that the same metadata elements were inspected as much in both catalogues. Surprisingly, it revealed that although the author and title received much less attention compared to the book's content and keywords, they were momentous predictors for book selection.

In traditional library, catalogues books and authors are treated as objects whose basic metadata are usually indexed. To incorporate diverse knowledge about fiction literature Mäkelä and his team had implemented *BookSampo Project* [9], a system which indexes not only bibliographical information but also focuses on content and context of text. The database employed functional content-centered indexing, ontological vocabularies and the networked data model of linked data [9]. The source of metadata for the adult fiction collection is HelMet Web Library, which is the web service of the city libraries of the Helsinki Metropolitan area in Finland. The Finnish fiction ontology *Kaunokki* is used for indexing the books in the online catalogue. *Sampo* caters to two functionalities - browsing and searching. The browsing interface allows the user to walk through the semantic network utilizing the book's actors and the keywords. Searching can be done using cover image or text. Interestingly, it claims to answer complex question like: "where in Finland are the most crimes committed in fiction literature ?" The system produced rich and diverse results compared to the ones produced by the traditional *Sata* catalogue [11].

Ross [22][21][23] and her students in one of the most compendious work, have done a study over Canadian adult leisure readers. They had interviewed 194 fervent readers to analyze how they selected books for pleasure reading. Book selection seemed to be a complex process. Readers chose books based on their mood. Book

elements like subject, character depiction and cover page influenced their selection apart from other factors. Ross's study were distinct than many others for the fact that it perceived that book selection is beyond just mere browsing a book collection or searching in a library database, but also had an affective parameter. Ross's study also attributed the role of family and friends in influencing book choices.

Apart from Ross, Pejtersen and Austin [18] and Smith [26] have given importance to the role of affect in influencing a reader in the book selection.

One of the interesting implementation for finding fiction using icons or pictures have been done in *BOOK HOUSE* [17], which is based on user-librarian conversations for finding fiction. A reference book is selected by the user after navigating through entities called 'facets'. The facets are access points to novels, representing various important attributes of the novel as perceived by the readers. Retrieval and ranking of the books with respect to the reference book is based on the similarity of the features, termed as dimensions, some of them being - plot, setting, place, time and genre. The evaluation was done in a public library and showed it was easy and gratifying to use. The users found it useful to select the book they intended especially when they are not sure of their interest. However, the features are dependent on the meta-data as keyed-in by the librarian and also the reader-librarian conversation. This poses fundamental challenge in scaling up the indexing process due to its manual nature. This might not capture well, the entire essence of the book, i.e. the features have not been derived by actual processing of the text of the fiction literature. With the increasing list of literature it becomes difficult to track the essence of all books, and users might not get serendipitous results.

Tang et al. [28] conducted a user study of aNobii, a social networking site designed for readers. It allows users to rate, review and discuss them with other readers. It provides three search functionalities : searching books by authors, similar bookshelves and friends' bookshelves. Browsing based on the meta-data - author's name was found to be the most efficient compared to the remaining two. However, the other two options produced more serendipitous choices.

3 Concept

3.1 Notion of Similarity in Text

The phrase 'notion of similarity in text' has a variety of meanings and interpretations; often depending on factors like *context* and *perspective*. When do we state that two text documents are similar? Broadly, we may classify it in three major ways. First, if there are many matching words in the two documents, which is popularly referred to as 'lexical similarity'. Second aspect is when there is similarity in the 'message' that is conveyed in the text (even though the words are not exactly the same). This 'message' may be interpreted as the 'meaning' of the text content, which is popularly classified as 'semantic similarity'. The third variant is 'syntactic' similarity which is more focused on the structural aspect of text, an example being sentence construction ways linguistic perspective. Often, there are overlaps between these notions, and there are hybrid approaches.

3.2 Vector Space Model

In the popular vector space model [24] of text retrieval, we are primarily leveraging 'lexical similarity' while searching for text. For example, if we search with *Return of Sherlock Holmes* as query book, Lucene is expected to fetch books that has a lot of overlapping words, which may be common words like 'murder', 'crime', 'police' for a detective genre. Although the order of words are not important, but this technique is highly successful in many practical scenarios. The model is primarily 'matching' words and accumulating scores for each matching word to arrive at final relative rank. Under the hood, we create a vector from each document, where each word of a document represents a dimension of the vector. The dimensions are represented by weights. Similarity between documents are calculated as the cosine similarity between vectors (which may run from hundred to thousand of dimensions). Apache Lucene (we have used version 6.0) [1] is an implementation of vector space model, with more advanced modifications (like Okapi BM25) to derive the weights for each term. We use Lucene model in what is popularly known as *bag-of-words* (BOW) setting, where the order of the words lose their meaning while calculating the similarity.

[1] https://lucene.apache.org/core/6_0_0/core/

3.3 SIMFIC Model

We specify SIMFIC's notion of similarity here. We attempt to address the semantic and syntactic similarity between texts. As it happens mostly in fiction text, there are similarity between books based on multiple aspects (features) like writing style, sentiment, plot complexity and others. For example, while searching for a book similar to *Emma* by Austen, the reader may not only be interested to retrieve books which have huge matching words. We can conjecture that an author will have a certain vocabulary and indeed *Emma* and *Pride and Prejudice* indeed share a lot of words, authored by same person and are hence quite similar(Lucene is expected to perform and pick similar items this way). But how about retrieving *Miss Marpel Series* by Agatha Cristie while searching for *Emma*? Both of them share a certain common writing style, emphasis on female gender but from different authors. Literary critics often label both as 'female oriented', and hence they are similar in this aspect. We expect SIMFIC to address such information needs based on semantic similarity on literary aspects.

Books are relatively longer documents. We express each book as a collection of smaller units called 'chunks'. We have extracted numeric feature vectors to represent each chunk. Similarity is computed as the inverse of Euclidean distance between chunk feature vectors (refer equation 3). We have experimented with L2, L1 norm, cosine similarity and decide on L2. The important question that arises, is to decide the size of a chunk and there is no one answer to this. There is some guidance in literature [8], on selection of a 'good' chunk size, which can range from ten to fifteen thousand words. We inspected our corpus and found that the smallest book is around ten thousand and the longest one is more than a two hundred thousand words. We selected the chunk size as ten thousand words (details in the following section). The chunks are of same size for the entire corpus. After calculating normalized similarity values between chunks, we use an aggregation model to accumulate the final similarity between books. We call this model as SIMFIC (Similarity in Fiction).

3.4 Compare and Contrast: SIMFIC Versus Vector Space Model (BOW)

There is an inherent similarity between both models based on the fact that, both arrive at similarity as similarity between numeric vectors. We draw out a few fundamental differences between both models, on how each define their notion of 'similarity'. First, vector space model in 'BOW' setting is an evidence accumulation engine, whereby documents (books here) similar to the query document are retrieved and ranked based on the presence and absence of words in the text. Presence of a word (which is a feature) is rewarded with weights (tF), along with a trade off for rewarding rare words and penalizing common words (IdF). In SIMFIC, books are retrieved and ranked based on the similarity between feature vectors that capture the notion of similarity. These vectors are based on 22 low level content based features, which we feel capture the literary notion of similarity in fiction text. Hence, books are not retrieved based on accumulating and rewarding keyword match between documents or meta-data search but based on the combination of the literary features like - "writing style, sentence complexity, sentiment of the plot, ease of readability, female oriented, male oriented, plot complexity and lexical richness, rural or urban setting". Secondly, we have performed feature (column) normalization in SIMFIC; whereas in the standard vector space model instances (rows) are normalized. Thirdly, SIMFIC calculates similarity at various levels of a book (chunks), and further accumulate these similarity weights to a book level (with penalty based on length). Lucene on the other hand has a single 'level' of calculating similarity. Finally, we can conclude the comparison by stating that Lucene is based on lexical similarity while SIMFIC is based on semantic and syntactic similarity.

3.5 Choice of features

There are twenty two low level features. Twenty of them are generated for each chunk and two of them for the book. The features generated for each chunk are - Flesh reading score, sentiment (positive, negative, neutral), presence of parts of speech (coordinating and subordinating conjunction, interjection, preposition, pronouns - possessive, personal, male, female), occurrence of punctuation like (quotes, hyphen, semi-colon, colon, period, comma, ellipses), average sentence length and paragraph count. The book level features are - number of characters in the story and Type Token Ratio (TTR). It should be noted that the definition of features mentioned here are qualitative in nature. Hence their definition in most cases are fuzzy and even critics differ with various schools of thought. Hence, we have made simplifying assumptions. The reason for choosing the aforementioned features and how they have contributed to the conceptualization of the high level features (referred in the Table 1) have been explained below.

Table 1: Features

Group	Feature Type	Feature
1	Writing Style	Paragraph count (f0), Female Pronoun (f1), Male Pronoun (f2), Personal Pronoun (f3), Possessive Pronoun (f4), Preposition (f5), Colon (f9), Semi colon (f10), Hyphen (f11), Interjection (f12), Punctuation and sub-ordinating Conjunction (f13), Sentence Length (f14)
2	Sentence Complexity	Co-ordinating Conjunction (f6), Comma (f7), Period (f8), Punctuation and sub-ordinating Conjunction (f13), Sentence Length (f14)
3	Female oriented	Female Pronoun (f1)
4	Male oriented	Male Pronoun (f2)
5	Rural or Urban Setting:	Quotes (f15), Number of characters (f20)
6	Sentiment:	Negative (f16), Positive (f17), Neutral (f18)
7	Ease of readability:	Flesch Reading Score (f19)
8	Plot complexity:	Number of characters (f20)
9	Lexical richness:	Type Token Ratio (f21)

Writing Style : Readers search a book based on an author because they might have liked the author's writing style or the subject on which he writes. Hence, writing style plays an important role in capturing the user need.

Authors have a unique style of writing termed as 'tics'. More or less authors fret over their pattern of writing. After a day spent working on two sentences of *Ulysses*, James Joyce is reported to have said: "I have the words already. What I am seeking is the perfect order of words in the sentence" [2]. We might deduce the writing style from the usage pattern of parts-of-speech. A comparison between the use of personal pronouns among authors also brings out the difference in their writing style or subject. The use of punctuation like ellipse and dash attributes to it as well. Ellipsis suggest faltering or fragmented speech accompanied by confusion, insecurity, distress, or uncertainty. While dashes jolt you forward, ellipses make you pause and linger [15]. James Joyce has a tendency of using dash to start a dialogue and ellipses to communicate a pause (as in *Ulysses*), but D.H Lawrence does not. Average sentence along with paragraph count also contributes to the writing pattern.

Most of the authors scribe around certain subjects which has a close relation to the genre [8]. The use of prepositions and punctuations like comma seem to be useful in distinguishing certain genres. *Bildungsroman* is a literary genre that focuses on the psychological and moral growth of the protagonist from youth to adulthood, in which character change is extremely important. The genre under utilize words like - *upon, by, this* compared to other genres [8]. Comma is over utilized compared to period in historical novels leading to long sentences. Exclamation mark is heavily used in *New-gate* novels which comprises writings thought to glamorize the lives of criminals they portrayed. The use of locative prepositions along with other features could be used to distinguish *Gothic* novels from others [8], as the genre being heavily "place oriented". The findings above reveals strong evidence that different part of speech and the punctuations and their usage can help uncover the genre leading to uncovering of writing style.

Female or Male oriented : Readers might want to read plots having a male or female protagonist, or a story revolving around females or males. The use of pronouns [19] along with other features can be used to derive high level features - whether the plot is female oriented (having more female pronouns) or male oriented (having more male pronouns). Readers having read novels penned by Herman Melville and Jane Austen will often encounter that both authors write about different subjects and each is having a unique signal or style. Austen who writes primarily about women is far more likely to use female pronoun compared to Melville. Also the more obvious thing is the absence of many women characters in *Moby Dick*. The use of such subtle features - articles, conjunctions and the like could be used in revealing author's individual style [19].

Sentence Complexity : Sentence complexity can have various connotations. For example, readers who had read and liked *Great Expectations* by Dickens, might like the works by Edgar Allan Poe, as they write long sentences with lot of punctuations and coordinating conjunctions. Readers might want to select a book having similar sentence complexity with respect to the query book. For example, children might prefer reading novels which have lucid sentence construction.

Ease of readability : Though readers might seek books similar to writing style or sentence complexity, they might also want to read books based on similar reading easability. Reading-ease test gives an idea of the level of /empeasiness in reading the text. Readers might prefer to read a book as - 'easy-to-read' as the query book. There are various measures known as reading scores that quantifies the ease of reading text. We have used one popular measure - Flesch reading score [4].

Sentiment : Sentiment plays an important role in selection of a book. Most of the readers browse through the first and last few pages of the book when they do not have much idea of the book [27] ,in order to understand the sentiment flow in the plot. This sometimes influences them in the selection of the fiction. Sentiment assessment was done at chunk level as sentiment over the entire book would not have revealed anything sensible or interesting, as most of the books start with a different sentiment and it changes throughout the plot until the end. Comparison of sentiment among the chunks of two books would help in understanding the similarity between two books with respect to the flow of sentiment throughout the entire story.

Plot Complexity : Plot complexity is an important attribute in fiction text, often a book having multiple sub-plots intertwined. We have made a simplistic approach wherein a plot is complex if it has a lot of characters. Russian author Leo Tolstoy's *War and Peace* is regarded as one of the central work of world literature. The novel harbors at least twenty primary characters attributing positively to the complexity of plot. Hence, in-spite of the book being one of the finest piece of literature, some readers may wish to avoid, as it involves a very complex plot.

Rural or Urban Setting : Readers often select books based on the subject of the story, apart from writing style and other factors. The subject is often influenced by the urban and rural setting. We have made a simplistic assumption that this setting is governed by the amount of conversation and the number of people in the story, using the concept from the influential work of the Russian critic *Mikhail Bakhtin*.

Lexical Richness : Lexical richness may be defined as the degree of usage of distinct words in the text. It measures the quality of vocabulary in a language sample. Lexical richness can be used to compare the concordance of two authors [8]. A reader might choose a book based on lexical richness in order to enhance his vocabulary, specially the new learners or children learning a new language. Though lexical richness can be considered to be a subset of writing style it has been intentionally kept as a sole feature to account for the usage of unique words.

Excluded Features : Meta data like author's name and genre were initially considered and omitted as it did not help in getting diverse results. Same goes for frequent words, as they were introducing bias in certain genre like - detective and mystery, sea-faring, adventure and romantic ones specifically. This seems to occur because of obvious use of words like 'murder, police' in detective literature and 'love' in romantic novels to name a few.

4 Preprocessing and Feature Extraction

4.1 Corpus

Project Gutenberg website acts as our source repository. 19th century English novels and short stories authored by single person have been considered. Short stories authored by multiple authors have been omitted, as it was challenging to automate the extraction and segregation of content authored by multiple authors. Poems and essays were also excluded. Since, each author would have different style of writing and the stories might be based on different subject. This might be considered in our future work. Finally, we have 996 books that we consider as our corpus. The books are in epub format with additional meta-data like author's name, title, genre, year of publication and others.

4.2 Data Cleaning

Books in epub format from the *Gutenberg* repository are being parsed into HTML format using Epublib [2] Java API and further preprocessing done is mentioned below.

It was observed that the actual content was always followed by the text "*START OF THIS PROJECT GUTENBERG*". Hence, any content before the aforementioned text was eliminated. The lines starting with the text such as- *"project gutenberg, gutenberg ebook, all rights reserved, transcribed from.."* were removed, as they contained Gutenberg related meta-data and not the actual content. Author's name and book's title were also removed as it was irrelevant.

It was further observed that the actual content was present inside the *p* tags. Typically the headers and table-of-contents were inside the *h* or the *anchor* tags. Hence, most of the times the content inside the *h* and *anchor* tags could be used to detect and remove the irrelevant text from the main content.*p* tags with certain style class attributes contained irrelevant text like - bibliography, index and headers. These were eliminated from the actual content. Most of the books' actual content seemed to end with the signature text *"End of the Project Gutenberg EBook"*. Hence, any text after the aforementioned text was removed. However, some books contained advertisements of other books, footnotes and bibliography before the signature text, which is irrelevant. So, a further check was performed.

Stanford NLP API (version 3.8.0) [3] was leveraged for the various preprocessing activities like tokenization and lemmatization. Typical stop word removal process was not adopted (except for the 'TTR' feature), since many of the stop words were features (example: prepositions, pronouns). Lemmatization was performed, stemming was not performed. Two versions of the content was maintained, one is the raw content (on which sentiment and certain features were extracted) and the other being lemmatised content for another set of feature extraction. This treatment varied from feature to feature depending on the need. For TTR, we had to remove both stop words and punctuation, since here we count the unique words. In this case, we have a special list of stop words for fiction text which includes the 'usual' English stop words and character names. The characters names were derived based on certain simplifying assumptions (see *Plot Complexity* below) and recommendations based on experiments by Jockers [8].

4.3 Chunking

Each fiction text has been divided into sections or chunks. Two chunking strategies have been implemented. In one strategy each chunk comprises 10,000 words. The size was determined after experimenting with various numbers between 5,000 to 20,000. We checked the effect of chunk size on TTR calculation. It was found that 10,000 chunk size provided TTR values which could be prominently differentiated over known novels. Jockers also found 10,000 to be a good choice for 19th century fiction and Shakespeare [8]. Since, last chunk will contain less than 10,000 most of the times, it was appended with the text from the first chunk in a circular fashion, so that it contains exactly 10,000 words. This we call as 'circular chunking'. In the second strategy chunk size is being taken as 2000, and the last chunk is not appended with the text from the first chunk.

4.4 Features

The following twenty features have been extracted for each chunk of the book - "paragraph count, female pronoun, male pronoun, personal pronoun, possessive pronoun, preposition, colon, semi-colon, hyphen, interjection, punctuation and sub-ordination conjunction, coordinating conjunction, comma, period, double-quotes, sentiment (positive, negative, neutral), sentence length, Flesch Reading Score". The remaining two features - "number of characters in the story and Type Token Ratio (TTR)" have been calculated at the book level. Stanford NLP API has been used to derive the Part-of-Speech (using part of speech annotation), identify character names (using named entity tag annotation), assess sentiment of the text (using sentiment annotated tree), and derive lemma for each sentence for each chunk.

The following high level features referred in the table 1 have been derived from the low level features mentioned above.

[2] http://www.siegmann.nl/static/epublib/apidocs/
[3] https://nlp.stanford.edu/nlp/javadoc/javanlp/

Ease of readability : The high level feature comprises only one low level feature - Flesch reading-ease score (FRES). This reading score has been used in our model to find the ease with each a text could be read. Higher scores in the Flesch reading-ease test indicates that the text material that is easier to read. A score range of 50-70 implies text readable by school children (8th-12th grade) . The score range 30-50 implies readable by college going students and is termed as 'difficult to read', and finally the range 0-30 implies for college graduates and is termed as 'Very difficult to read. Best understood by university graduates. ' The formula considered for the same is given as:

$$FRES = 206.835 - 1.015(TotalWords/TotalSentences) - 84.6(TotalSyllables/TotalWords) \quad (1)$$

Sentiment : The feature comprises three low level features - positive , neutral and negative sentiment. Sentiment has been assessed for 10% of the text for each chunk by performing random sampling over sentences in the chunk. The evaluation was done on random samples as it was time and resource consuming to perform analysis on the entire text, especially when the sentences were complex, i.e. having lot of punctuations and coordinating conjunctions. Each chunk can have one of the sentiments based on the score calculated by the Stanford NLP package - positive (score 3-4), neutral (score 2) and negative (score 0-1).

Writing Style : Writing style feature is a combination of the following twelve low level features - average paragraph ,the occurrence of female pronoun, male pronoun, personal pronoun, possessive pronoun, locative preposition and words signifying activity around a location, colon, semi colon, hyphen, interjection, punctuation and sub-ordinating Conjunction and median Sentence Length.

Lexical Richness : The feature comprises only one feature - TTR. It is a measure of the ratio of unique words to total words in a given text. *Moby Dick* authored by Herman Melville has a TTR of 8.2 and has a large vocabulary of 17,072 unique words compared to Sense and Sensibility by Jane Austen, having TTR of 5.2 and 6,315 unique words when chunking done over 10,000 random samples. Therefore readers will encounter more unique words in *Moby Dick* compared to Jane Austen's text. The low level feature is highly influenced by the length of novel. If TTR is done over the entire book, then longer books will have less TTR compared to shorter ones. Hence, to overcome this problem either we can take random samples of fixed number of words or append text to the last chunk (as the last chunk will have less words). We have chosen the latter approach and implemented it as circular chunking to calculate TTR.

Plot Complexity : The feature - number of characters is being utilized to roughly understand the 'complexity of plot'. After the text processing the character names are derived with simplifying assumptions. Considering a character name as - *John*, can be referred to in the story by multiple names - *(John , Doe, John Doe, Mr. John , Mr. Doe, J. Doe , Johny, Johny Doe)* etc. In such cases we derive the final name as Johny Doe and counting it as one character. But, in cases having same first name but different surname as - *(M. Prosper, M. Ferdinand)* we keep them as different characters. This method is simple but cannot detect different nick names of the same person.

Rural or Urban Setting : The features - the number of characters and the occurrence of quotes present per chunk has been utilized to ascertain the high level feature - rural or urban setting.

4.5 Distribution of Features

We plotted the distribution of features and wanted to make some investigation on the basic statistical distribution of the features. Figure 1 displays the distribution. There are some features that approximately follow the bell curve of a normal distribution. These features are f0, f1, f3, f4, f5, f6, f7, f8, f13, f18, which are paragraph count, female pronoun ratio, personal pronoun ratio, possessive pronoun ratio, preposition ratio, conjunction ratio, comma ratio, period ratio, conjunction-punctuation ratio and neutral sentiment ratio respectively. Of course, all of them have different mean and standard deviation, but they exhibit a

'normal' trend. Some of the features (f12 - interjections) seem to follow log normal. Some features that have a distinctly different shapes are: f9 (colon ratio), f10 (semicolon ratio), f11 (hyphen ratio). Flesch reading score f19, has a skew towards higher values. We have performed feature scaling and mean normalization, with a fairly standard technique as shown in the formula 2. The entire feature extraction is performed as an one time activity which takes approximately eighteen hours for our corpus of 996 books (on a laptop with Intel Pentium i5 processor and sixteen gigabyte memory).

$$FeatureNorm = (FeatureValue - FeatureMean)/(FeatureMax - FeatureMin) \tag{2}$$

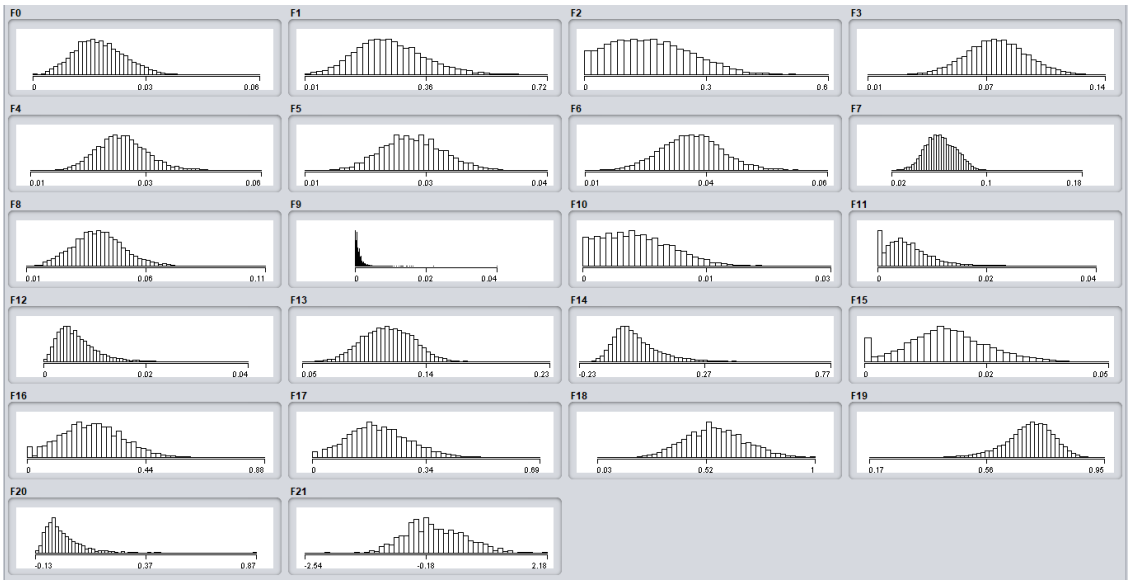

Fig. 1: Distribution of Features

5 Building the Similarity Model

5.1 Steps in building the Model

Each chunk is represented by a feature vector. The basic idea is to calculate similarity between query book chunks and the corpus chunks. We accumulate the similarity weights from chunk to book level. We penalize very long books during the accumulation process and finally create a rank relevance list. We use the algorithm 1 to build the model, which has been explained below.

For each chunk of the query book, we calculate the L2-similarity with all other chunks of the entire corpus (refer equations 4, 5 and 3), and store the interim results. Next, we iterate over the result of the last step. For each key, we check if there is a matching chunk of corpus, that is present in any other list. If a matching chunk of corpus is present, we 'roll-up' the values. This roll up from chunk level to book level, of the query book can happen by addition (refer equation 6) or multiplication (refer equation 8). Since the L2 similarity values computed before is normalized, product of numbers less than one produces an even smaller number. So, we go for aggregation by addition. Besides that, in order to decide for addition, we experimented with a small corpus of around forty 'known popular' books like *Great Expectations*, *Pride and Prejudice*, *Return of Sherlock Holmes*. We had an approximately fair idea of the expected results, given a query book; Example, *Great Expectation* and *Oliver Twist* are both authored by Dickens, with similar writing style, genre and mood. This is fairly corroborated unanimously by English fiction experts to a good extent. We iterate twice over the result of previous step to accumulate weights and find similar books from the corpus, given the query book. This list is sorted in decreasing order to get the final relevance rank.

Algorithm 1 SIMFIC Model - Pseudo Code to return relevant books based on a query book

procedure SIMFIC(FV, QB) ▷ Input: Feature Vectors (FV) and Query Book (QB)
  for <Loop Over all chunks of query Book QB> **do**
    <Find L2 Similarity between QB chunks and corpus FV, return RESULTS1>
    <RESULTS1: Key = QB(i), Val = List of (Weight, Similar Chunk) >
  end for
  for <Loop Over RESULTS1> **do**
    <Aggregate similarity weights per corpus chunk, return RESULTS2>
    <RESULTS2: Key = CorpusChunkId, Val = Aggregate Weight>
  end for
  for <Loop Over RESULTS2> **do**
    <Aggregate weights per book of similar corpus chunk, hold in RESULTS3>
    <Penalize weights by Number of chunks, update RESULTS3 >
    <RESULTS3: Key = SimilarBookId, Val = Penalized Weight >
  end for
  for <Loop Over RESULTS3> **do**
    <Sort and Normalize, return SORTEDRESULTS>
    <SORTEDRESULTS: Key = Normalized Weight, Val = Corpus Book Name>
  end for
  return $SORTEDRESULTS$ ▷ Output: Sorted Relevance Rank List of Books
end procedure

5.2 Penalty factor

The final sorted list is based on aggregation of similarity weights. However, there seems to be a problem in this aggregation method. Often the longer books were popping up in the result list inspite of it being not that similar compared to other books in the list. Very long books have more chunks compared to shorter ones. In some cases, even though a long book's chunks are average similar to the query book chunks, many of their chunks add up and start appearing similar by mere aggregation of similarity values. Hence, came the need to penalize such long books. We experimented and found that dividing the total similarity value (which has now exceeded one, after aggregation) by a factor like 'number of chunks' (refer equation 6) or by a smaller factor like square root as shown in 7) are certain ways of optimization. Finally, we selected 'number of chunks' based on experiments. As a result, very long books like *David Copperfield* by Dickens, which was initially appearing for any random query book in the top ten search results, is now heavily pushed down in the ranking, far away from top ten results.

However, there is a flip side to the above technique. Since we penalised the 'long factor' we encountered another unintended problem. A shorter book having less chunks and average similar to the query book chunks were pushed up because of the penalisation.

$$L2Similarity(between\ chunks) = \frac{1}{1 + L2Distance(between\ feature\ vectors)} \tag{3}$$

$$Q(product\ space) = (Chunks\ Of\ QueryBook) * (All\ Chunks\ Of\ A\ Given\ Book) \tag{4}$$

$$P(corpus\ space) = (Number\ Of\ Books) \tag{5}$$

$$AggregateSimilarity\{Penalized\ by\ chunks\} = \frac{(\sum_{j=1}^{P}(\sum_{i=1}^{Q} L2Similarity))}{N} \tag{6}$$

$$AggregateSimilarity\{Penalized\ by\ SqrRoot\ of\ chunks\} = \frac{(\sum_{j=1}^{P}(\sum_{i=1}^{Q} L2Similarity))}{\sqrt{N}} \tag{7}$$

$$AggregateSimilarity\{By\ Product\} = \frac{(\prod_{j=1}^{P}(\prod_{i=1}^{Q} L2Similarity))}{N} \tag{8}$$

5.3 How does the model behave on 'known books'?

Before having the formal user study, we experimented on known popular books, and we found promising results. Example, when we take *Pride and Prejudice* as a query book, the top five results are *Emma*, *Mansfield Park*, *Sense and Sensibility*, all by Jane Austen. SIMFIC results are quite promising since they have a female protagonist and have similar writing style. It might be interesting to note, that for the same query book, in a bag-of-words vector space model using Lucene, the top five results returned also contain *Emma*, *Mansfield Park*, *Sense and Sensibility*. We discovered similar trends for another classic query book: *Great Expectations* by Dickens. A very similar book by Dickens : *David Copperfield* appears in top two, for both SIMFIC and Lucene. We do not claim that above results having books by same author as the query book is testimony of a good model. The results seem to be of similar writing style and genre. Indeed there are some results which need deeper literary analysis. We will discuss these aspects in more detail, when we present the evaluation of the results by user study. The preliminary results encourages us, to stop extracting more features and proceed to design a user study to evaluate the effectiveness of SIMFIC compared to a Lucene baseline.

5.4 Feature Selection

We applied feature selection (FS) technique to the SIMFIC search results. We wanted to address the problem, wherein we empower the users with a short and simple explanation on why the search results are retrieved by the system, for the given user query. The motivation being, that many text retrieval systems seldom 'explain' their results. This is often a case of frustration for user, specially in cases, where we do not have keyword or direct meta data (author, genre) match. A book is a classic example of such a use case. We attempted to address this problem, exploiting the various feature selection methods. There is however a limitation to this approach. Feature selection reduces the overall performance, since for each query there is a costly statistical computation.
However, when we compare SIMFIC with Lucene, we find that Lucene also consumes a decent time for processing each query. The average query processing time is about five to ten seconds, in a setting where a whole book is parsed as a query. This turned out to be an 'equalizing effect' during the user study, since SIMFIC and Lucene both consume almost similar time to retrieve results, thereby reducing or eliminating any bias due to system performance.

Steps on calculating the feature selection:

1. Given a query book, we calculate a ranked list of top ten books. We label these books (represented by a single global feature vector per book) to a class one. All other books of the corpus are labeled as class zero. So now we have a labeled dataset.
2. FS can be solved in many ways. Two prominent ways are treating FS as a random search problem over feature space with a classifier to evaluate the effectiveness of the feature sub set. Another method is to exploit some measure like information gain, gain ratio to discriminate classes.
3. We experimented with both techniques. Since the first method with classification is much more costlier than the second method, we selected the gain ratio based method (using Weka Java API version 3.8.1 [4]) to select top three to four attributes out of twenty two features. However, our experiments with classifier fetched promising results, where we found average classification accuracy of more than 85 percent with just three to four features. The top three attributes are displayed as 'important factors' to the user in SIMFIC.

6 Evaluation and Results

6.1 User Study Design

The user study design was done with the purpose of evaluating the effectiveness of SIMFIC in fiction book search. We intend to present same front-end (refer Figure 2) but different back-ends to the users. There are three systems:

[4] https://www.cs.waikato.ac.nz/ml/weka/index.html

1. SIMFIC system with the feature selection (FS) 'System Earth'
2. SIMFIC system 'System Saturn'
3. Lucene based system 'System Lunar'

We present these systems in Latin Square design to minimize bias due to display order (see Figure 3, which shows different display order) and such related things. There were three main sections in the user study:

1. Section one dealing with user demographics, feature importance perception by user
2. Section two with a two fixed 'known popular books' that the user is familiar with
3. Section three dealing with a book, selected by the user, assuming that the user is familiar with the book

In section two, we present the three systems in Latin Block design. In section three, we use System Earth only.

6.2 User Study setting

The user study was held in a classroom. Users were given a very brief introduction of the system and informed of the basic steps to be completed for the tasks. Users accessed the application that was hosted in each of the laptops, there was no network involved. Users were provided with printed copies of the questionnaires (a copy is attached in appendix). On average, each user required a time of about twenty to thirty minutes to complete all tasks on the laptop and manually enter the responses in the questionnaire.

6.3 Participant demographic details

Twenty participants - 18 students (bachelor and master students) and 2 professors of English, Jadavpur University (Kolkata, India) participated in the user study. The users can be deemed to be subject matter experts of English fiction. Most users have expressed their familiarity in using search engines for fiction text.

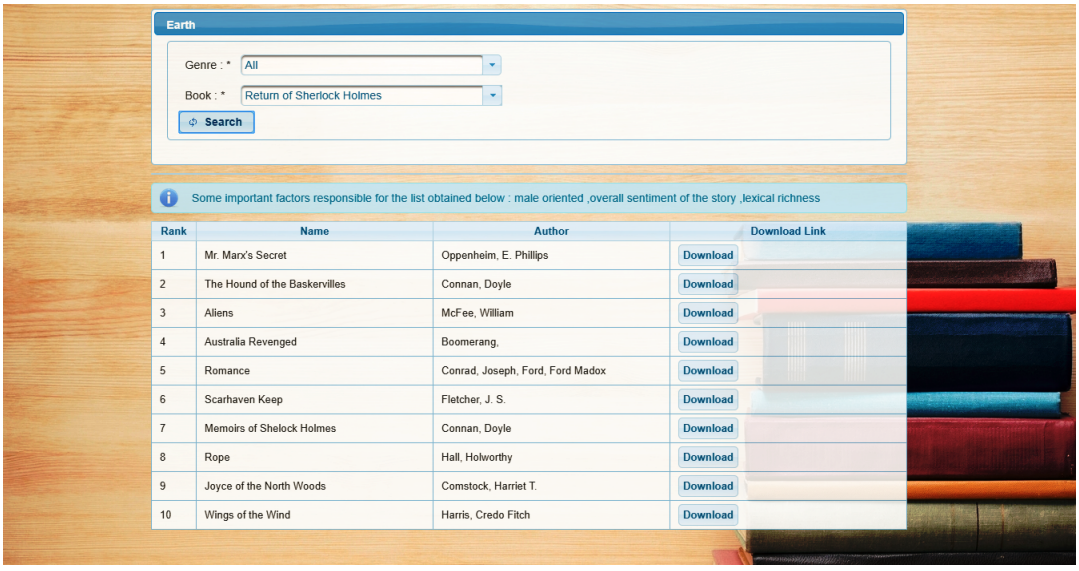

Fig. 2: Search Web Interface - Same for all three systems. Saturn and Lunar only lack the explanation portion, shown here for Earth

	1	2	3
1	A	C	B
2	B	A	C
3	C	B	A

Fig. 3: Latin Block User Study Design : A = Saturn, B = Earth, C = Lunar

6.4 Faculty background

Given the fact, that Kolkata was the former capital city of British ruled India, the English department of the institute has a track record of attracting top talent from the entire country for English research and education. The faculty is a nurturing ground for many writers, luminaries in the field of English and humanities for over past six decades. The department [5] is a partner in the Leverhulme Trust International Network Project on Commodities and Culture in the Colonial World with Kings College London, New York University, and the University of Technology, Sydney. 19th century English fiction is a research 'Thrust Area' of the department. We consulted the head of the English department, and selected two 'known popular books' as: *Hard Times* by Charles Dickens and *Pride and Prejudice* By Jane Austen. These two classic books are taught in detail as part of the program. So we can assume that these books are good baseline data points to compare SIMFIC and Lucene. We understand that there might be an implicit bias by the participants since they are taught these books, but we argue that having our model evaluated by 'trained' participants is a better choice on a whole, than having random participants to perform the same.

6.5 Results: Analysis of user demographics and feature perception

We present the results of the user study in this section. Table 2 shows the user demographic details. We extracted twenty two features. We grouped these features in 'layman terms' as broad categories like 'Ease of readability' , 'Writing style'. We did not perform any particular feature weighting in the implementation, however we wanted to get an insight to find the importance of each of these features of a scale of one to five (we are not doing a ranking amongst features). We present the statistics in Table 3. Following are the salient inferences drawn from the evidence of Table 2 and 3:

1. We have a major section of the participants, who are young bachelor and master students of English literature, with the average age being 25. Besides, there are two professors of age 52 and 72, which are outliers by age, but add heavily to the overall quality and diversity of our participant dataset.
2. Most users have expressed their familiarity in using search engines in general, with a very high daily usage of web search and propensity to buy books on-line. The evidence being a mean value of 4.75 out of 5, with regard to frequency of using web search. A fair evidence of buying books on-line also exists (mean 2.65 out of 3), thereby we can deduce that the cohort is well versed in searching books in an on-line web browser or mobile app setting.
3. The top three features which are attributed by the users as top factors when searching for fiction books are: Writing style (mean = 4.05 out of 5), Ease of readability (mean = 3.60), Overall sentiment of the book (mean = 3.85). These evidence defend our initial design assumptions, when we made a choice of

[5] www.jaduniv.edu.in/view_department.php?deptid=67

features. We focused quite heavily on quantifying 'writing style' - which is a very qualitative aspect in literature. This is supported by over ten features in SIMFIC. Ease of readability is supported by a fairly accepted measure in the community - Flesch reading scores (also available as a feature in the SIMFIC back-end). Overall sentiment of the book is supported by three features - positive, negative and neural sentiment, all of them adding up to one. However we understand that there are definitely a host of other features revolving in the minds of fiction experts, which is beyond SIMFIC. They are captured in the section two and three of the study.

Table 2: User Demographic Details

Item	Statistics	Additional Details
Age	Mean = 25, Median = 21, St.Dev=12	Includes two professors aged 52 and 72
English Qualification	BA (85%), MA (30%), Prof. Dr. (10%), Others (5%)	–
Mother Tongue	Bengali (55%), English (5%), Sindhi (5%), Malayalam (5%)	–
Frequency of Web Search	Mean = **4.75**, St.Dev=0.55	On a 1-2-3-4-5 likert scale, 5 highest
Frequency of Buying Books On line	Mean = **2.50**, St.Dev=0.69	On a 1-2-3 likert scale, 3 highest

Table 3: Feature perception by Users

Fiction Functional Aspect	Statistics	Related Features in SIMFIC ?
Ease of readability	Mean = **3.60**, St.Dev = 0.99	Yes - One
Writing Style	Mean = **4.05**, St.Dev = 0.83	Yes - Sixteen
Overall Sentiment	Mean = **3.85**, St.Dev = 1.18	Yes - three
Author name	Mean = 2.80, St.Dev = 1.20	None
Author period	Mean = 3.15, St.Dev = 1.18	None
Author gender	Mean = 1.75, St.Dev = 1.10	Yes - One
Genre	Mean = 3.50, St.Dev = 1.10	None
Vocabulary Size	Mean = 2.85, St.Dev = 1.31	Yes - One

6.6 Comparison of search effectiveness: SIMFIC Versus Lucene

The results of effectiveness of SIMFIC and its comparison with Lucene given a 'known popular book' query is captured in the second section of the user study. The results of effectiveness of SIMFIC when using a familiar query book selected by a user is presented in Table 4. Following are the salient inferences:

1. How does SIMFIC perform when its search effectiveness is compared to Lucene BOW model?
   We have attempted to answer this question by asking this question implicitly and explicitly in a Latin block design. We have posed the question in a likert scale (higher values are better) of one to five: "Are the search results helpful?" The question is repeated for 'know popular books' as well as 'user selected book'. Evidence of table 4 suggest that for both known books *Pride and Prejudice* and *Hard Times*, SIMFIC has a mean helpfulness value of 4.15 (against the 3.45 by Lucene) and 3.80 (against 3.40 by Lucene), respectively. The standard deviation helpfulness values are also higher for Lucene.
2. Since we may have a user bias since System Earth (SIMFIC with FS) provides feature selection explanation as against Lucene without an explanation, we turn to the bare SIMFIC which does not provide explanation (System Saturn). We find that SIMFIC is more helpful based on mean helpfulness value compared to Lucene. For *Pride and Prejudice* and *Hard Times*, SIMFIC has a mean helpfulness value of 3.75 (against the 3.45 by Lucene) and 3.55 (against 3.40 by Lucene), respectively.

3. When explicitly voting for the best system (by testing the systems in random order and asking :'Amongst all three systems - Lunar, Earth, Saturn, which one provides the best results, given *Hard Times* and *Pride and Prejudice* ?'), we find that SIMFIC is a clear winner, with 12 users opting for System Earth (SIMFIC with FS), 3 users voting for System Saturn (SIMFIC), and 5 voting for System Lunar (Lucene), refer Figure 4.

4. Besides the core question over helpfulness, we also posed the question of novelty of results ('Do you find any pleasant surprises?'), we find that Lucene is a slightly better performer in comparison to SIMFIC, over known books. Evidence over all twenty participants on *Hard Times* reveal that the mean value on 1-to-5 likert scale, is 3.25 for SIMFIC, 3.3 for SIMFIC with explanation, 3.55 for Lucene. The standard deviation for SIMFIC being 1.37, SIMFIC with explanation as 0.98, and Lucene has 1.32. In light of standard deviation values, we may infer that serendipity is almost uniform across systems. This perhaps shows in general, the randomness of text retrieval systems.

5. Users were asked to input in three or four sentences, what they think are good features in deciding a similar book. This question was posed, after user searched a book ('known popular' and 'user choice'). We received open answers to such an open question. After the user study, we realized that this might be a point of potential bias, since we also provide explanation based on feature selection, which might in turn be reproduced by the users. On analysis, we present few top factors mentioned by users by rephrasing certain points without diluting the actual essence of the statement mentioned by the user. We have merged similar aspects and brought up the distinct ones in table 5. Some features are relatively challenging to quantify and extract, for example, how to quantify 'society'?.

Table 4: SIMFIC Versus Lucene: Using 'Hard Times' and 'Pride and Prejudice (5 being highest)

Books/Systems	SIMFIC	SIMFIC with FS	Lucene
Avg. Helpfulness (with Std. Dev) for *Pride and Prejudice*	3.75 (1.07)	**4.15 (0.67)**	3.45 (1.35)
Avg. Helpfulness (with Std. Dev) for *Hard Times*	3.55 (1.05)	**3.80 (0.95)**	3.40 (1.09)

Table 5: Comparison of Top Features, Query Book = *Pride and Prejudice*

Features	Feature Selection by SIMFIC	Top Features as per User
1	Writing Style	Period and Times of Author
2	Lexical Richness	Storyline
3	Ease of Readability	Society
4	Sentence Complexity	Genre and Theme

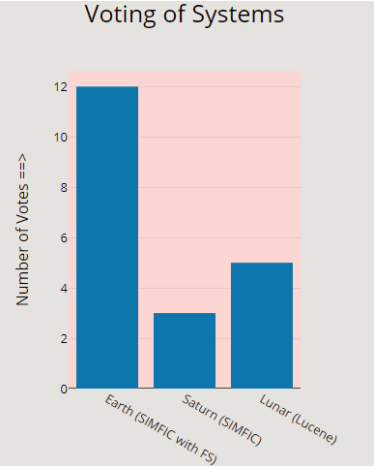

Fig. 4: Voting of Systems

6.7 A qualitative comparison beyond statistics

We present few observations on the behavior and a qualitative comparison of SIMFIC and Lucene (Table 6). We feel these results are interesting and worthy of a literary analysis. We explore into a specific genre and compare the systems. We select 'Detective Genre' for SIMFIC, and select a popular book: *Memoirs of Sherlock Holmes* by Doyle. We find the top ten relevant books as per SIMFIC contains five detective books, and two of them are works of Conan Doyle. To draw the attention at this junction, when we retrieve similar books in SIMFIC, we do not have any 'filter' of searching for similar objects within only the selected genre or by same author, as the query book. This indicates, that our choice of features are performing decently well to retrieve matching books. For the same book now when we consider the Lucene results: we find that three books authored by Doyle are retrieved, which will be somewhat obvious due to word by word matching in a high dimensional vector space [24]. The results by Lucene are perhaps a bit more predictable, SIMFIC has some surprises. For example, the top item of SIMFIC is a thriller *Mr. Marx's Secret* by Phillips Oppenheim. We further explored to see if this behavior is repeated for other genres. On using literary genre, for the query book being *Hard Times*, the second result was a pleasant surprise (Table 7). It is an English translation of a German book *Problematische Naturen*, the translation named *Problematic Characters: A Novel* by Friedrich Spielhagen. This book is similar to *Hard Times* on many aspects. For example both were published almost within few years from each other: *Hard Times* in 1854 and *Problematic Characters: A Novel* in 1848. Both represent the troubled times in the European society during a revolution of those times. The themes of the book are quite similar. But we do not have insights into the writing style of the authors and treat this as a novel result. For the query book being *Pride and Prejudice* the comparison is available in Table 8.

Table 6: Comparison of Top Results, Query Book = *Memoirs of Sherlock Holmes* By **Connan, Doyle**

Rank	SIMFIC Result	Lucene Result
1	*Mr. Marx's Secret* By Oppenheim, E. Phillips	*Return of Sherlock Holmes* By **Connan, Doyle**
2	*Tales of Terror and Mystery* By **Connan, Doyle**	*The Hound of the Baskervilles* By **Connan, Doyle**
3	*Captivating Mary Carstairs* By Harrison, Henry Sydnor	*The Sign of Four* By **Connan, Doyle**
4	*Return of Sherlock Holmes* By **Connan, Doyle**	*One Of Them* By Lever, Charles James
5	*A Captain in the Ranks: A Romance of Affairs* By Eggleston, George Cary	*Alone* By Harland, Marion

Table 7: Comparison of Top Results, Query Book = *Hard Times* By **Dickens, Charles**

Rank	SIMFIC Result	Lucene Result
1	*David Copperfield* By **Dickens, Charles**	*David Copperfield* By **Dickens, Charles**
2	*Problematic Characters: A Novel* By Spielhagen, Friedrich	*Out of a Labyrinth* By Lynch, Lawrence L.
3	*Oliver Twist* By **Dickens, Charles**	*Deadham Hard: A Romance* By Malet, Lucas
4	*Sunrise* By Black, William	*The Invisible Lodge* By Paul, Jean
5	*Mabel's Mistake* By Stephens, Ann S	*The Complete Prose Works* By Tupper, Martin Farquhar

Table 8: Comparison of Top Results, Query Book = *Pride and Prejudice* By **Austen, Jane**

Rank	SIMFIC Result	Lucene Result
1	*Emma* By **Austen, Jane**	*Munster Village* By Hamilton, Lady Mary
2	*Mansfield Park* By **Austen, Jane**	*Emma* By **Austen, Jane**
3	*Sense and Sensibility* By **Austen, Jane**	*Deadham Hard: A Romance* By Malet, Lucas
4	*Barren Honour: A Novel* By Lawrence, George A.	*Sense and Sensibility* By **Austen, Jane**
5	*The Road to Mandalay - A Tale of Burma* By Croker, B. M.	*Mansfield Park* By **Austen, Jane**

6.8 Open suggestions and Overall helpfulness of SIMFIC

Overall high level question as well as open suggestions and criticism on improvement of the prototype is mentioned here. We have the following inferences based on that evidence of Figure 4:

A Users were asked, if they liked system Earth overall. Indeed, it is debatable on the effectiveness of this question: 'Was the system Earth helpful? Yes or No?'. We find that 16 out of 20 participants (80%) voted as 'yes'. This question might be futile, since Lucene might perhaps also get a similar response. But in a different light, evaluation of the usefulness of SIMFIC for fiction retrieval system receive votes of 80% of the participants. At this juncture, an interesting observation: the two professors differed on this question. One of them voted 'yes' for SIMFIC, the other voted as 'no'. The person who voted 'no' liked the Lucene based system.

B Open suggestions, likes and dislikes were captured. The top ones are reproduced below, by rephrasing certain terms, so that we could merge similar criticism without loss of meaning:

1. We need a bigger corpus. Many students expected certain books that are part of their study program but not available as part of our corpus. They wanted to check how the system behaves with such query points but could not.

2. The major *Likes*, *Dislikes* and *Suggestions* of the system without any ranking (we could ask this question in general because of a uniform UI) are:

   (a) *Likes:* 'user friendliness', 'the ranking system', 'element of serendipity', 'explanation of search results', 'relevant results'
   (b) *Dislikes:* 'the explanation was often incomplete or inadequate', 'not all 19th century novels present', 'short summary of books absent', 'the criteria for selecting a book is not always clear'
   (c) *Suggestions:* 'include more books', 'include fiction from other languages', 'a brief overview of each novel', 'more details of search results like date of publication, movies if any, created from the book or its adaptation'

7 Limitations of current work and future research

We discussed a plethora of features that we feel can help in searching fiction books. Of course this list is not exhaustive. There is an obvious scope of improvements and additions possible to the current list of features.

Another area of improvement is using a more efficient and optimized machine learning method of feature selection that is more faster. On the user study aspect, having more data is many a times a may lead to a practical inferences, than drawing inference from a small set of users. Having more number of participants with an improved eye tracker technique might also uncover other aspects. More participants might also enable making effective use of statistical hypothesis testing on the final voting of results to compare systems.

There might be another radical view point to learn the features implicitly using deep learning methods. To the best of our knowledge, there is no reported work making use of deep learning on fiction text corpus. There is one clear challenge of directly making use of publicly available Google word-to-vector trained vectors, since they are trained over news corpus (or product reviews from Amazon), So it may not be a very good idea to directly lift the trained vectors and use them to deduce similarity. It sounds interesting to learn word vectors from the *Gutenberg* corpus and check if the fascinating observations like 'Germany minus Berlin' = 'France minus Paris', do hold in the fiction 'word2vec' [12] space. Can we have such revelations over books? or over chunks of books?

On the UI aspect, there are some immediate scope of improvement. We have received one very important suggestion during the user study. It is about a provision in the system to show a summary of the books, that appear in the search result. We did not allow web access to the user during the user study. Some users also complained on this aspect. It might also be a good idea to use borrow such summary of books from Wikipedia dump. However this is inherently a data preprocessing challenge, since not all digitized books on *Project Gutenberg* have a Wiki summary. Having a Google search text box placeholder within with our prototype and observing the user behavior during search may also uncover interesting aspects of the user behavior, while searching fiction books. However we feel that studies on a user search journey are perhaps a slightly different research topic.

8 Conclusion

We have presented SIMFIC, a similarity model for content based search of fiction text. The model is based on hand crafted features which is expected to match human perception of similar fiction books. We implemented a simple UI for the given model. An exactly same UI was also created with Lucene as back end. We compared the systems in a user study by subject matter experts. We found primary evidence that the model is helpful in addressing the fiction search needs for the given corpus. There are some limitations. However we do understand that the claims regarding SIMFIC being more effective compared to Lucene is perhaps not statistically significant given a relatively low number of participants (twenty). Statistical hypothesis test with a low number of data points are a perhaps not an effective way to differentiate between the null and alternative hypothesis. But we argue that the participants are trained English fiction experts who are 'taught' these books in classroom as a lecture. We have a fair basic evidence that SIMFIC using a novel similarity mechanism on a small feature space is a good competitor to Lucene, which has a bag-of-words methodology to find similar documents in a high dimensional vector space.

Acknowledgment

The authors would like to thank the students and faculty members of the Department of English, Jadavpur University, Kolkata, India for their cordial support in conducting the user study in March 2018.

References

1. Denice Adkins and Jenny E Bossaller. Fiction access points across computer-mediated book information sources: A comparison of online bookstores, reader advisory databases, and public library catalogs. *Library & Information Science Research*, 29(3):354–368, 2007.
2. Frank Budgen and Hugh Kenner. *James Joyce and the making of Ulysses*. Indiana University Press Bloomington, 1960.

3. Josephine Davidson and Roderick Cave. *Fiction in Wellington: a survey of user preferences in public libraries in greater Wellington*. Victoria University of Wellington, Department of Librarianship, 1990.
4. Rudolf Franz Flesch. *How to write plain English: A book for lawyers and consumers*. Harpercollins, 1979.
5. FOBID Netherlands library forum. The Netherlands Library Statistics 2015, July 2016.
6. Brad Gauder. Perceptions of libraries, 2010: Context and community. a report to the oclc membership. *OCLC Online Computer Library Center, Inc.*, 2011.
7. Deborah Goodall. *Browsing in public libraries*. Library and Information Statistics Unit LISU, 1989.
8. Matthew L. Jockers. *Macroanalysis: Digital Methods and Literary History*. University of Illinois Press, Champaign, IL, USA, 1st edition, 2013.
9. Eetu Mäkelä, Kaisa Hypén, and Eero Hyvönen. Improving fiction literature access by linked open data-based collaborative knowledge storage-the booksampo project. In *78th IFLA General Conference and Assembly, Helsinki*, 2012.
10. Anna Mikkonen and Pertti Vakkari. Readers' search strategies for accessing books in public libraries. In *Proceedings of the 4th Information Interaction in Context Symposium*, pages 214–223. ACM, 2012.
11. Anna Mikkonen and Pertti Vakkari. Finding fiction: search moves and success in two online catalogs. *Library & Information Science Research*, 38(1):60–68, 2016.
12. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
13. Ministry of Education and Culture of Finland. Library Statistics Finland 2017, April 2017.
14. National Endowment for the Arts. How a nation engages with art: Highlights from the 2012 survey of public participation in the arts. National Endowment for the Arts Research Report no 57, September 2013.
15. University of Chicago Press Staff. *The Chicago Manual of Style*. Univ. of Chicago Press, 17th edition, 2017.
16. Suvi Oksanen and Pertti Vakkari. Emphasis on examining results in fiction searches contributes to finding good novels. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, pages 199–202. ACM, 2012.
17. Annelise Mark Pejtersen. A library system for information retrieval based on a cognitive task analysis and supported by an icon-based interface. In *ACM Sigir Forum*, volume 23, pages 40–47. ACM, 1989.
18. Annelise Mark Pejtersen and Jutta Austin. Fiction retrieval: Experimental design and evaluation of a search system based on users'value criteria (part 1). *Journal of documentation*, 39(4):230–246, 1983.
19. James W Pennebaker. The secret life of pronouns. *New Scientist*, 211(2828):42–45, 2011.
20. Janna Pöntinen and Pertti Vakkari. Selecting fiction in library catalogs: A gaze tracking study. In Trond Aalberg, Christos Papatheodorou, Milena Dobreva, Giannis Tsakonas, and Charles J. Farrugia, editors, *Research and Advanced Technology for Digital Libraries*, pages 72–83, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
21. Catherine Ross et al. Finding without seeking: what readers say about the role of pleasure reading as a source of information. *Australasian Public Libraries and Information Services*, 13(2):72, 2000.
22. Catherine Sheldrick Ross. Finding without seeking: the information encounter in the context of reading for pleasure. *Information Processing & Management*, 35(6):783–799, 1999.
23. Catherine Sheldrick Ross. What we know from readers about the experience of reading. *The Readers' Advisor's Companion*, pages 77–96, 2001.
24. Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.
25. S Serola, P Vakkari, S Serola, and P Vakkari. The public library in the activities of people. *Finnish. Publications of the Ministry of Culture and Education*, 21, 2011.
26. Duncan Smith. One reader reading: A case study. *Guiding the reader to the next book*, pages 45–70, 1996.
27. David Spiller. The provision of fiction for public libraries. *Journal of librarianship*, 12(4):238–266, 1980.
28. Muh-Chyun Tang, Yi-Jin Sie, and Pei-Hang Ting. Evaluating books finding tools on social media: A case study of anobii. *Information Processing & Management*, 50(1):54–68, 2014.
29. The Reading Agency (2013). Library facts, March 2014.

User Study on 19th Century English Fiction Books

The user study consists of four parts.

- The 1st part consists of a basic questionnaire.
- In the 2nd part, the user searches for similar books given a "known popular book".
- In the 3rd part, the user selects a book of their choice from the 19th Century digitised books and evaluate the quality of the results.
- The 4th part is general feedback about the tool.

The results of the study will only be used for research. The storage and evaluation of these results will be anonymous.

Age: _____ years **Gender**: ☐ female ☐ male

Part 1: Basic Questionnaire (5 minutes)

- What is your native language?

  ☐ Bengali ☐ English ☐ Hindi ☐ Other: _____

- How would you rate your proficiency in reading and understanding English?

  Basic skills ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Expert / Mother tongue

- Educational Qualification

  ☐ Student-BA ☐ Student- MA ☐ PhD-English/Humanities
  ☐ Professor - English/Humanities ☐ Others_____

- How frequently are you using web search engines (e.g. Google, Bing)?

  ☐ Never ☐ Monthly ☐ Weekly ☐ Daily, once ☐ Daily, several hours

- How frequently do you buy books online (e.g. from Amazon, Flipkart.com)?

  ☐ Never ☐ Occasional - Once or twice a year ☐ Regular Buyer

- When researching a English Fiction book, what existing tools and websites are you using?

  ☐ Google.com ☐ Google Scholar

  ☐ 'Project Gutenberg' website ☐ 'uread.com' book website

  ☐ Any special eBook repository, specify:

  _____

- When searching for similar English fiction books to "a given book" , the results might be debatable. Therefore, most of the results might seem irrelevant in some cases or a good matching book may be somewhere down in the list or if you are lucky, the best matching book might just be at the top!

  When you are evaluating a similar book, how important are the following parameters :

	Not important				**Important**
Ease of readability ?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
Writing style?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
Overall sentiment of the book ?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
The author name?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
The author period ?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
The author gender ?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
The genre?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5
The size of vocabulary ?	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5

Part 2.A (6 mins.)

Select **System A** from browser bookmark.

Task 1 :

- From drop-down/combo box, select **Genre - "All"**
- Now select **Book -"Pride and Prejudice"** (by Jane Austen)
- click on **Search** button.

Note the search results. The best matched book is the first in the list, followed by books in decreasing order of relevance.

- Please judge the **helpfulness** of the result list with respect to writing style, overall sentiment of the book, ease of readability.

  Not helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful

- Which four books do you think are most similar to "**Pride and Prejudice"**? (your opinion can differ from system results)
  1) _____
  2) _____
  3) _____
  4) _____

- Which top salient aspects of the text, do you hold responsible for the choice of similar books above?
    1) _____
    2) _____
    3) _____
    4) _____

Task 2:

- From drop-down/combo box, select **Genre - "All"**
- Now select **Book -"Hard Times"** (by Charles Dickens)
- click on **Search** button.

Note the search results. The best matched book is the first in the list, followed by books in decreasing order of relevance.

- Please judge the **helpfulness** of the result list with respect to writing style, overall sentiment of the book, ease of readability.

  Not helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful

- Which four books do you think are most similar to " **Hard Times "**? (your opinion can differ from system results)
    1. _____
    2. _____
    3. _____
    4. _____

- Which top salient aspects of the text, do you hold responsible for the choice of similar books above?
    1) _____
    2) _____
    3) _____
    4) _____

Part 2.B (4 mins)

Select **System B** from browser bookmark.

Task 1 :

- From drop-down/combo box, select **Genre - "All"**
- Now select **Book -"Pride and Prejudice"** (by Jane Austen)
- click on **Search** button.

Note the search results. The best matched book is the first in the list, followed by books in decreasing order of relevance.

- Please judge the **helpfulness** of the result list with respect to writing style, overall sentiment of the book, ease of readability.

Not helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful

- Note a short **explanation** of the results has been provided. Is that helpful?

  Not Helpful ☐ ☐ Helpful

Task 2:

- From drop-down/combo box, select **Genre - "All"**
- Now select **Book -"Hard Times"** (by Charles Dickens)
- click on **Search** button.

Note the search results. The best matched book is the first in the list, followed by books in decreasing order of relevance.

- Please judge the **Helpfulness** of the result list with respect to writing style, overall sentiment of the book, ease of readability.

  Not helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful

- Note a short **explanation** of the results has been provided. Is that helpful?

  Not Helpful ☐ ☐ Helpful

Comparing the current and the previous system, which one provides better results?

☐ Current ☐ Previous

Part 2.C: (4 mins)

Select **System C** from browser bookmark.

Task 1 :

- From drop-down/combo box, select **Genre - "All"**
- Now select **Book -"Pride and Prejudice"** (by Jane Austen)
- Click on **Search** button.

Note the search results. The best matched book is the first in the list, followed by books in decreasing order of relevance.

- Please judge the **helpfulness** of the result list with respect to writing style, overall sentiment of the book, ease of readability.

  Not Helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful

Task 2 :

- From drop-down/combo box, select **Genre - "All"**
- Now select **Book -" Hard Times "** (by Charles Dickens)
- Click on **Search** button.

Note the search results. The best matched book is the first in the list, followed by books in decreasing order of relevance.

- Please judge the **helpfulness** of the result list with respect to writing style, overall sentiment of the book, ease of readability.

  Not Helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful

Comparing the current and the previous system, which one provides better results?

☐ Current ☐ Previous

Part 2.D: (2 mins)

Comparison of all **systems A, B and C**

- Amongst all three systems, which one provides the **best results,** given two popular books: "Hard Times" and "Pride and Prejudice"?

  ☐ A ☐ B ☐ C

Part 3: 'Go-as-you-like' with the research tool. (3 minutes)

You can search any book from any genre in order to retrieve search results.

We have considered these aspects to find similar books, given a book:

- Writing Style
- Sentiment - positive, negative or neutral mood of the book
- Ease of readability, Vocabulary Richness

Please use the given research tool to answer the following question, by selecting **System B** from browser bookmarks.

Select a **Genre**. Now select a **Book** (Select one that **you are familiar with** and the book is neither "Hard Times" nor "Pride and Prejudice"). Click on **search** button.

- Selected Book Name:_____
- Top four books **expected by you** (may differ from books retrieved in search results)
  1) _____
  2) _____
  3) _____
  4) _____
- Helpfulness of results:

  Not Helpful ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 Helpful
- Note a short **explanation** of the results has been provided this time. Is that helpful?

  Not Helpful ☐ ☐ Helpful

Part 4: Feedback about the research tool. (2 minutes)

You can provide comments (without being specific about systems A, B, or C) in general about fiction search engine.

- Did you find the tool helpful for fiction text search purpose?

  Not Helpful ☐ ☐ Helpful

- What features of the tool did you **like and dislike** the most?

  Like: _____

  Dislike: _____

- Any other suggestions for the fiction search engine?