SIMFIC: An Explainable Book Search Companion

***Abstract*—Consider a digital library of fiction books. A user has a certain book in mind and is searching for books which are similar in writing style, sentiment and general content. Classic retrieval techniques applied in such scenarios lack the support to explain why a set of top-K items are relevant to the query. Explainable AI (XAI) is attempting to explain the working mechanism of complex models like retrieval systems, helping humans to trust systems as companions. XAI research suggests two prominent directions: either develop add-on methods to peek inside complex models or design simple and explainable models. We adopt the latter and present a simple and explainable model for fiction books called SIMFIC ("similarity in fiction"). We partition each book into smaller portions called chunks. We extract features aligned with human cognition like writing style, sentence complexity, sentiment per chunk. In a query by example setting, a relevance ranked list of books is created based on similarities between chunks. A novel reward-penalty scheme is used while accumulating the similarities to ensure a fair comparison between short and long books. We perform feature selection using global feature vectors and pose them as plausible explanations, arguing them as global key factors differentiating between relevant and non-relevant books. SIMFIC is compared with a benchmark retrieval model and evaluated by domain experts in a study. Majority of the users found SIMFIC to provide more helpful results with respect to writing style, sentiment and general content, compared to the baseline. The results are statistically significant.**

***Index Terms*—explainable AI (XAI), text retrieval, fiction, digital companion**

I. Introduction

Data driven methods have created new benchmark performance across domains driven by the availability of abundant data and advancement of computing power. The information retrieval community has also seen a surge of complex learning models applied in design of ranking [1], text and query representation [2]. It is plausible that more advancement will be made by such complex models. However, this impressive performance comes with an added cost of dealing with complex black boxes. These models are often not explainable and hence might not be trusted by human beings due to absence of transparency. An interdisciplinary field of research is active in recent times dedicated to this transparency factor, often referred to as the explainable AI (XAI [3]).

In a classification setting, XAI can help to understand why a certain prediction was made for a query instance [4], which may be due to some model parameters or features. Explanations have different connotations depending on the perspective of the user (application user, data scientist or regulator). In a retrieval setting, often the primary goal is to know why a certain document is relevant to the given query [5]. Often this is driven by the notion of similarity encoded by the designer in the model. The challenge lies in "opening up" this notion of similarity to the end user.

Consider we are searching for books similar to a particular book. We often cannot express the information need only in terms of simple keywords or topics but on the whole "content" of the book, thus requiring other methods like querying by example. In this work we extract semantic and syntactic features for shorter "chunks" of a book; the motivation is to exploit similarities at various sections of a book during retrieval. We accumulate the similarities with a novel reward-penalty trade-off scheme to fairly compare a short and a long book, this helps to retrieve shorter books which may be similar in writing style, content when compared to a long query book. This work has two contributions. Firstly, SIMFIC is built ground-up using "literary features" based on a popular XAI strategy of building a simple and understandable model from scratch. The model enables capturing similarity at various parts of a long document. Secondly, we provide features as plausible explanation of retrieving a set of top-K results, evaluated by domain experts in a user study. Link to the system and the code is provided[1].

II. Background

We present a snapshot of related work across content based information retrieval (CBIR), digital humanities and XAI. CBIR have been governed by handcrafted features with success stories in images and video [6] using features such as color, shape, textures, edges with suitable transformations. In text retrieval, there have been works on exploiting topics modelling. Given that each document of a corpus is a generated from a set of topics with varied proportions, a ranked list of documents can be retrieved for a given query document, if they share similar topics [7].

In the context of content based search in digital humanities and library science, major efforts [12] are focused on gathering rich metadata of books, across languages. Often in practice, the meta-data is employed to simply "filter" the corpus by author name, genre, publication year, language and then explore the corpus. Digital Humanities projects in English and German (*Visualizing English Print*[2] and *Digital Rosetta Stone*[3]) developed tools like metadata tagger, topic models, morphosyntactic annotations and visualization to support retrieval tasks.

Recent works on explainability in AI, in a classification setting has various add-on methods specific to ANNs like Layer wise relevance propagation [4], model agnostic methods like Local interpretable model agnostic explanations (LIME)

[1]https://github.com/obfuscatedforreview/

[2]http://graphics.cs.wisc.edu/WP/vep/

[3]http://www.dh.uni-leipzig.de/wo/drs/

and Causality based methods like independent conditional expectation [8]. In a retrieval setting, there are much less contributions on the XAI aspect. In [9] authors try to provide a "global answer" in a learning to ranking setting. In this setting the authors used a base black-box ranker to create secondary training data to learn a new interpretable model. On another work [5] named Explainable Search (EXS), attempt was made to adapt the LIME technique for a retrieval setting. In EXS, the user selects a particular search result and clicking an "explain" button shows the list of words contributing to this result.

III. Method

A. Notion of Similarity in Text

The notion of "similarity in text" has a variety of interpretations, often depending on the perspective. Broadly, we may classify it in three major ways. *(i)* "lexical similarity" refers to situations if there are many matching words in two documents. *(ii)* "semantic similarity" refers to situations when there is similarity in the "content" that is conveyed in a text, even though the words or the sentence are not exactly the same. *(iii)* "syntactic similarity" is focused on the structural aspects of text like grammar, part-of-speech and sentence structure.

B. Vector Space Model vs SIMFIC

In the classic Vector Space Model (VSM [10]) of text retrieval, we are primarily leveraging "lexical similarity" while searching for text. Open source software Apache Lucene[4] is a practical implementation of the VSM. In this work, VSM and Lucene is used interchangeably. VSM is primarily "matching" tokens and accumulating evidence for each matching token between query terms and the corpus. Under the hood, VSM uses a sparse vector representation of each document, where each word of a document represents a dimension of the vector. Similarity between documents is calculated as the cosine similarity between vectors. Although VSM treats content as a bag-of-words losing context of text, it has been a popular and practically effective text retrieval model. Hence VSM is selected as a baseline. In SIMFIC, books are retrieved and ranked based on the similarity between relatively dense and compact content based feature vectors that capture the notion of similarity. To conclude, VSM is based on lexical similarity while SIMFIC is based on semantic and syntactic similarity.

C. Choice of Literary Features: Why and How?

Readers often search a book based on their favorite author's writing style or topic or their own mood. A comparison between the use of personal pronouns, punctuation like ellipse and dash often brings out the difference in their writing style or subject. Ellipsis suggest faltering or fragmented speech accompanied by confusion or uncertainty. The author 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. The use of prepositions and

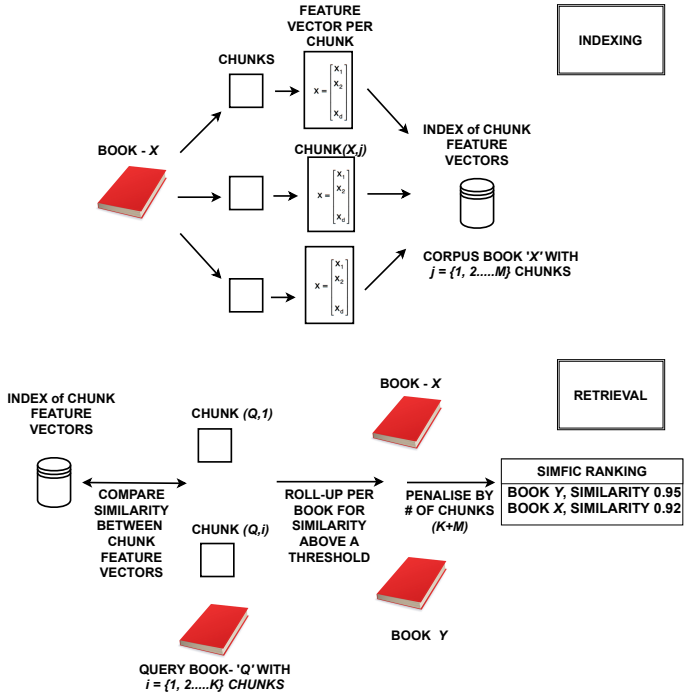

Fig. 1. Indexing and Retrieval

punctuation like comma seem to be useful in distinguishing certain genres. Comma is over utilized compared to period in historical novels leading to long sentences [11]. The use of locative prepositions along with other features could be used to distinguish *Gothic* novels [11]. We capture writing style by a broad feature category comprising twelve low level features like function words, punctuation, sentence length, paragraph count and others. On similar assumptions based on prominent literary factors [11] [12] and common consensus in the 19th century English fiction community, we selected the remaining literary features (Table 1). The reasons for selecting the remaining literary features are escaped for brevity. Around 1000 novels and short stories belonging to the 19th century English fiction, from the Gutenberg project[5] is our corpus.

D. Indexing and Retrieval using SIMFIC

The first step is to create chunks of a book. For each chunk we extract feature vectors (top Part, Fig 1). Consider a query book Q with K chunks (bottom part, Fig 1). We compute all possible similarities with indexed chunks and consider candidates only when the similarity is above a threshold. Next, we aggregate the similarity values per book and finally penalize the score per matching corpus book by total number of chunks. Similarity weights are sorted to get the final ranked list. Consider $Ch_{Q,i}$ as i-th chunk (feature vector) of query Book Q with K chunks and $Ch_{X,j}$ as j-th chunk of a corpus Book X with M chunks. Then the similarity between Q and

[4]http://https://lucene.apache.org/

[5]https://www.gutenberg.org/

X is given in (1), considering the sum over values when the similarity exceeds a threshold.

$$sim(Book_Q, Book_X) = \frac{\sum_{i=1}^{K}\sum_{j=1}^{M}\frac{1}{1+L2(Ch_{Q,i},Ch_{X,j})}}{K+M} \quad (1)$$

The novelty of this method is to fairly compare short and long books using the penalty factor. It "rewards" while accumulating similarities only when it exceeds a threshold and also "penalizes" many matches (long books). There are four parameters which were empirically estimated by experiments on a sample of 50 relatively well known books, where we have prior knowledge on similarity. Chunk size was estimated as 10,000 words [11]. The selected the similarity measure was *L2* norm (out of *L2*, *L1*, *cosine*); the penalty factor was selected as number of chunks *K+M* (out of *K+M*, *K.M*,$\sqrt{K+M}$, $\log$ *K.M*). The similarity threshold was fixed at *0.6* (out of *0.5*, *0.6*, *0.75*). Global feature vectors used for feature selection are computed per book by averaging the chunk vectors.

TABLE I
LIST OF FEATURES

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), Sentence Length (f14), Punctuation sub-ordinating Conjunction (f13)
2. Sentence Complexity	Co-ordinating Conjunction (f6), Comma (f7), Period (f8), Punctuation & 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)

E. Explainability using Feature Selection

We apply feature selection (FS) techniques to SIMFIC search results. Given a query book, we calculate a ranked list of top twenty books. We label this set of books as class one (*relevant*) and all others books of the corpus are labeled as class zero (*irrelevant*). FS can be solved in many ways. Two prominent ways are, either treating FS as a random search problem over feature space with a classifier performance to evaluate the effectiveness of a smaller feature sub set, or exploit some measure like information gain, gain ratio to determine association between features and classes. We experimented with both techniques and found similar results. Classification performance (using SVMs with RBF kernels) was in the range of 85% to 90% with a subset of three features. We argue that these features are plausible global "explanation" for the retrieved set because they can fairly discriminate relevant from non-relevant instances. Since the classification method is much more costlier, we used the gain ratio based method to select the top three features that are displayed in the UI as explanations (Fig 2).

IV. EVALUATION

There is no ground truth available stating the rank of similar books for a given fiction book of our corpus, hence we perform user studies for evaluation. SIMFIC was embedded in a simple prototype (see Fig.1 for UI) with focus on evaluating the back end in a user study. There are three systems with the same UI: SIMFIC with Feature Selection (FS) is called "Earth", SIMFIC without FS is called "Saturn" and Lucene is named "Lunar". Twenty users consisting of 18 students and 2 professors of a University's department of English[6] participated in the study. We assume them to be subject matter experts of English fiction. We selected two "known popular books", *Hard Times* by Dickens and *Pride and Prejudice* By Austen, which are taught as part of the lecture on English fiction. Following are the salient inferences drawn based on the results of the study:

1) Users search for books that are similar to two "known books" as query (mentioned above). Then we pose the question, "Please judge the helpfulness of the result list with respect to writing style, overall sentiment, ease of readability", on a 1-5 Likert scale for each system. Evidence suggest (Table II) that for both known books SIMFIC has a better mean helpfulness values (SIMFIC = 4.15, Lucene = 3.45, Query Book = *Pride and Prejudice*) with a comparatively lower standard deviation value for SIMFIC. The result is statistically significant with a p-value of 0.042 and $\alpha = 0.05$ in Wilcoxon Signed-Rank (WS) test.
2) There may be user bias as SIMFIC with FS provides explanation as against Lucene without an explanation. So we analyze Saturn (which has no explanation) and find that even bare SIMFIC has higher mean helpfulness value compared to Lucene (SIMFIC = 3.75, Lucene = 3.45, Table II).
3) When explicitly asked to select the best system, 60% users vote for System Earth (SIMFIC with FS), 15% users vote for System Saturn (SIMFIC), and 25% vote for System Lunar (Lucene). It was specifically asked, if the explanations were helpful in a binary manner. Majority voted in favour of SIMFIC, *e.g.* 85% and 70% users voted that SIMFIC explanations are helpful in explaining search results for the books *Pride and Prejudice* and *Hard Times* respectively.
4) On asking about the novelty of search results ("Do you find any pleasant surprises?"), we find that Lucene (3.55) is a slightly better performer in comparison to SIMFIC (3.33), over known books like *Hard Times*. However, Lucene's novelty performance score is not statistically significant in WS test (p-value = 0.336, $\alpha = 0.05$) .

V. CONCLUSION

Although the initial results are encouraging, there are limitations which directs us towards future research. The explain-

[6]www.UniversityNamePlaceObfuscated.edu

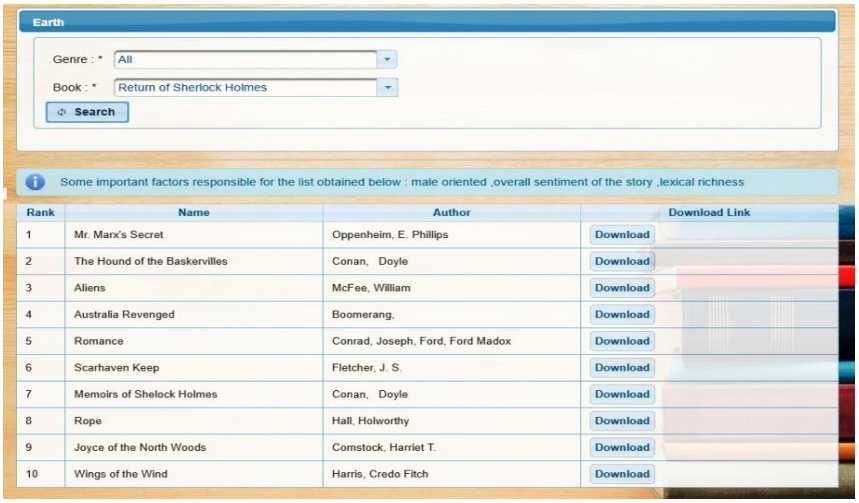

Fig. 2. UI with Query Book as *Return of Sherlock Holmes* and Explanations are marked as factors like *Male Oriented, Sentiment, Lexical Richness*

TABLE II
SIMFIC VERSUS LUCENE: USING *Hard Times* AND *Pride and Prejudice* (5 IS HIGHEST)

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

ability aspect can also be tackled by framing the problem in a linear regression setting, with top feature weights posing as explanation. The current study asked users if SIMFIC's explanations were helpful, it did not compare explanation between systems. We can attempt to make an objective comparison of explanation between both models; the UI for Lucene (VSM) could be revamped with simple term-frequency statistics of word matches as explanation. Does the method of representing fiction text by dense feature representations and using them for retrieval, also generalize well for other languages? Since search "helpfulness" is subjective, adding a system that retrieve books (pseudo) randomly, can be used as another baseline, to make the evaluation more objective. These options are currently being investigated. To conclude, we present a book search system focused on writing style, sentiment, general content. It is compared with the VSM as a baseline, for search helpfulness. It provides global explanations and is evaluated by domain experts with statistically significant results favouring SIMFIC. XAI driven research like SIMFIC will help AI systems to gain trust as digital companions.

REFERENCES

[1] J. Guo, Y. Fan, B. Croft "A deep relevance matching model for ad-hoc retrieval" [Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, 2016].
[2] B. Mitra, E.N alisnick, N. Craswell, R. Caruana "A dual embedding space model for document ranking" [ArXiv preprint arXiv:1602.01137, 2016].
[3] D. Gunning "Explainable artificial intelligence (xai)" [Defense Advanced Research Projects Agency (DARPA), nd Web, 2017].
[4] S. Bach, A.Binder, G.Montavon, K.Müller, S.Wojciech "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation" [Public Library of Science, 2015].
[5] J. Singh, A. Avishek "EXS: Explainable Search Using Local Model Agnostic Interpretability" [Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019].
[6] M. Flickner, H. Sawhney, W. Niblack, J. Ashley and others "Query by image and video content: The QBIC system" [IEEE Computer Society Journal vol.8, 1995].
[7] X. Yi, J. Allan "A comparative study of utilizing topic models for information retrieval" [European conference on IR, 2009].
[8] Q. Zhao, T. Hastie "Causal interpretations of black-box models" [Journal of Business & Economic Statistics, 2019].
[9] J. Singh, A. Avishek "Posthoc interpretability of learning to rank models using secondary training data" [ArXiv preprint arXiv:1806.11330, 2018].
[10] G. Salton, A. Wong, C. Yang "A vector space model for automatic indexing" [Communications of the ACM, vol.18, 1975].
[11] M. Jockers "Macroanalysis: Digital Methods and Literary History" [Book, University of Illinois Press, IL, 2013].
[12] A. Mikkonen, P. Vakkari "Readers' search strategies for accessing books in public libraries" [ACM Proceedings of the 4th Information Interaction in Context Symposium, 2012].