AN AGGRANDIZED FRAMEWORK FOR ENRICHING BOOK RECOMMENDATION SYSTEM

Tulasi Prasad Sariki[1*], G Bharadwaja Kumar[2]

[1,2]School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India

Email: tulasiprasad.sariki@vit.ac.in[1*](corresponding author), bharadwaja.kumar@vit.ac.in[2]

DOI: https://doi.org/10.22452/mjcs.vol35no2.2

ABSTRACT

In this era of information overload, Recommender Systems have become increasingly important to assist internet users in finding the right choice from umpteen numbers of choices. Especially, in the case of book recommender systems, suggesting an appropriate book by considering user preferences can increase the number of book readers in turn having an aftereffect on the users' lifestyle by reducing stress, stimulating imagination, improving vocabulary, and making readers smarter. The majority of book recommender systems in the literature have used Collaborative Filtering (CF) and Content-Based Filtering (CBF) methods. Even though CBF methods have shown better performance than CF methods, they are mostly confined to shallow linguistic features. The present work proposed an aggrandized framework having three concurrent modules to improve the recommendation process. NER module extracts the Named Entities from the entire book content which are the key semantic units in providing clues on the possible choices of reading other related books. The Visual feature extraction module analyzes the book front cover to detect objects and text on the cover as well as the description of the cover which can bestow a clue for the genre of that book. The Stylometry module enhances the feature set used in the literature to analyze the author's literary style for identifying similar authors to the present author of the book. These three modules conjointly improved the overall recommendation accuracy by 18% over the baseline CBF method that indicates the effectiveness of the present framework.

Keywords: Recommender Systems, Book Recommendation, Content-Based Filtering, Named Entities, Book Front Cover, Book Similarity, Natural Language Processing, Deep Learning

1.0 INTRODUCTION

Recommender System (RS) is an effective tool to address the problem of information overload which is unavoidable in many domains like e-commerce, advertisement, travel, entertainment, and social networking. For a better understanding of customer preferences, most of the retailers in the online environment are utilizing the RSs and gaining insights about their business. So, RS is not only advantageous for the consumers but also a boon for the retailers to maximize their profits and to reduce customer attrition. Due to these benefits, RSs have become part and parcel of our daily life. A good RS is always aimed at user-item interactions to provide meaningful recommendations. The user-item interactions are described in the form of a matrix and these interactions are explicit as well as implicit. Explicit interactions are usually mentioned in the form of ratings where these ratings can be numerical (at a point of scale on 1 to 10) as well as categorical (like or dislike). But sometimes these ratings include text comments also. Implicit interactions are like click, view, purchase, etc. Each kind of feedback has its advantages and disadvantages. The explicit feedback mechanism is very simple whereas the implicit feedback mechanism does not require the user to state his preferences like ratings or voting. The main motive of RSs is to utilize these user-item interactions and recommend the items based on user preferences. From a machine learning perspective, a RS problem will be treated as either a regression or classification task. If RS tries to predict the unknown ratings or preferences of a particular user on a point of the scale, then it's a regression problem. If it tries to predict unknown categories (like or dislike) of preferences or ratings, then it's a classification problem.

The problem of recommender systems has been studied comprehensively, and two main paradigms come into existence. Content-Based Filtering (CBF) recommendation systems attempt to recommend items similar to those a

current user has liked in the past, whereas Collaborative Filtering (CF) recommendation paradigm systems attempt to identify users whose preferences are similar to those of the current user and recommend items they have preferred. Most of the RS landscape is dominated by the CF models due to simplicity in their implementation, no domain knowledge requirement, and discovering the hidden preferences of the user. Despite these advantages, there are major drawbacks like cold-start, sparsity, scalability, gray-sheep users, shilling attack, long-tail effect, and lacking the way to include side information. In the CBF approach, the main challenge is to derive distinctive features from the items. Most of the CBF methods utilize hand-crafted item features for recommendations. The existing shortcomings of both CF and CBF are mitigated with other recommender models. Hybrid recommender models are introduced by combining both CBF and CF and other models. To mitigate the domain-specific challenges few other models like Demographic-based, Utility-based, and Knowledge-based models are developed.

It is a well-known fact that world-class online retailers and various service providers like Amazon, Netflix and Pandro realized the importance of recommendation engines to boost their sales. Not only the companies' academia also showed a vested interest in the recommendation field by publishing more papers and provided a separate conference named ACM Recommender Systems (RecSys) from 2007 on-wards. Even after all these developments, there are many domains of the recommendation process which have paid little focus or nothing. In line with the current statement, book recommendation is one such domain that requires much attention despite the works carried out in the literature.

Numerous studies and surveys indicated that there is a drastic diminution in book readership and reported the effect of book reading on the social behavior of the people. Hill and Capriotti et. al [1] is stating that there exists a clear relationship between book reading and positive involvement in society. Another statistical study from Hill Strategies Research Inc. Canada [2] stated that book reading will reduce stress, stimulate imagination, improve vocabulary and make readers smarter. If readers get proper book recommendations based on their preferences, there may be a chance of an increase in the number of book readers and will have a positive effect on the user's lifestyle. The launch of e-readers like the Kindle and the availability of books in digital formats make it much easier for the readers to get their interested books. This transformational shift should increase the number of book readers but the facts state that there is no upward trajectory in readership of e-books. At this juncture, a book recommender can play a paramount role in achieving this goal by recommending the right book at the right time.

Even though many book recommender systems have been proposed in the literature, most of them are very primitive in recommending books that can meet user choices and needs. They have used several advanced Artificial Intelligence approaches such as Content-Based, Collaborative Filtering, and Hybrid methods. Most of the existing Content-Based book recommenders used the handcrafted metadata features for analyzing the book content. The construction of such hand-crafted textual features is very complicated since natural languages are highly ambiguous. Also, exploiting full-textual content in the books for extracting relevant features is a very time-consuming and also non-trivial task. Hence, many works in literature have focused on only using certain portions of the text as well as the author's literary style. The present work tries to address this gap by proposing a framework to enhance the book recommendation process. The framework contains three independent modules where each module strengthens the book recommendation process by exploiting three unaddressed gaps in the literature. The first module utilizes a deep learning-based content analyzer to extract the Named Entities which are significant semantic units in understanding the book content and concepts from the full-text content. To the best of our knowledge, Named Entities have not been utilized in the book recommendation process in the literature. The second module applies effective deep learning algorithms to figure out the genre of the book from the book front cover and its description which has not been addressed in book recommendation literature. The third module enhances the Stylometry features used in the literature. Finally, all the modules are judiciously agglomerated to provide an effective framework in generating final recommendations.

The remainder of this article is organized as follows. Section-2 gives an account of literature work regarding book recommendation systems. The proposed aggrandized framework for book recommendation is described in Section-3. Section-4 depicts the experimental results and comparisons of baseline models. Section-5 gives conclusions of Aggrandized Framework for Enriching the Book Recommendation System.

2.0 LITERATURE REVIEW

There are only a few noticeable techniques are utilized in the book recommendation domain. The applied methods are purely depending on user preferences, hand-crafted metadata, and other information about the user like gender, age, location, etc. These works [3] tried to leverage the power of recommendation techniques applied to other domains like movies, e-resources, and products but, they didn't emphasize on special features of books.

The recommendation problem in the book domain has additional challenges as an individual's book reading preferences depend on many factors. Books are having typical characteristics than other text materials like news, movie plots, academic conferences or journal papers, and micro-blogs. Most of the recognized works in this domain have used CF models as well as CBF models. One of the popular online book recommender systems named goodreads.com [4] uses a hybrid model by fusing multiple proprietary algorithms. They majorly rely on user interactions and user shelf information. Amazon.com [5] is using a generic recommender system for books. Most of these popular book recommenders are not disclosing their recommendation algorithms to the public. There are only a few recommendation models that tried to combine both CF and CBF techniques. Table-1 contains the comprehensive list of state-of-art works carried out in the book recommendation domain.

Table 1: The comprehensive review of works carried out on book recommendation

S. No.	Paper Title	Techniques applied	Dataset Used
1.	Book recommender prototype based on author's writing style [6].	CF	LitRec dataset
2.	AI Powered Book Recommendation System [7].	CBF	Data scraped from www.goodreads.com
3.	Exploring author gender in book rating and recommendation [8].	CF	Book ratings dataset
4.	Hybrid Book Recommendation Engine [9].	CF and CBF	LitRec Dataset.
5.	CD-SPM: cross-domain book recommendation using sequential pattern mining and rule mining [10].	CF and Association rule mining	MovieLens and Book domain dataset.
6.	Recommending books for children based on collaborative and content-based approaches [11].	CF and CBF	Book-Crossing dataset from www.amazon.com
7.	College library personalized recommendation system based on hybrid recommendation algorithm [12].	CF and CBF	Data sets of Library of Inner Mongolia University.
8.	Book Recommender System Using Fuzzy Linguistic Quantifiers [13].	CF and Fuzzy Averaging	Indian top universities text books dataset.
9.	Design of Book Recommendation System Using Sentiment Analysis [14].	Sentiment Analysis, CF, KNN Algorithm	Amazon Book review dataset.
10.	Information Retrieval and Graph Analysis Approaches for Book Recommendation [15].	Information Retrieval models and Graph Analysis algorithms	Social book mark dataset.
11.	Group-based Latent Dirichlet Allocation (Group-LDA): Effective audience detection for books in online social media [16].	Group Latent Dirichlet Allocation	Data scraped from www.weibo.com and www.douban.com.
12.	CBRec: a book recommendation system for children using the matrix factorization and content-based filtering approaches [17].	Matrix Factorization CBF	Book-Crossing dataset
13.	A Book Recommendation method based on Paragraph Vector User's Book Arrangement [18].	Doc2Vec and CBF	Data scraped from CrowdWorks.
14.	Recommendation systems: Principles, methods and evaluation [19].	CF and CBF	Data from social networks.
15.	Social book search: the impact of the social web on book retrieval and Recommendation [20].	Information retrieval CF	Google scholar data
16.	Ask Me Any Rating: A Content-based Recommender System based on RNN [21].	CF and Deep Learning Models	Movielens and DBbook dataset
17.	Book Recommendation Platform using Deep Learning [22].	Deep Learning Models	Book Cover Images Scraped from Amazon

18.	Study of linguistic features incorporated in a literary book recommender system [23].	CBF and NLP Techniques	LitRec Dataset
19.	A survey of book recommender systems [24]	Various state of art techniques.	LitRec, and other datasets related to books.

3.0 METHODOLOGY

The proposed framework comprises three independent modules: Named Entity-based recommender (NER Module), Visual Feature-based recommender (Visual Feature Extraction Module), and Stylometry-based recommender (Stylometry Module). The NER module is intended to extract the named entities that are constituted in the book that are crucial in understanding the book content and concepts from the full-text content. The Visual Feature extraction module is geared towards understanding the correlations that exist in between the book front cover and its genre. The Stylometry module aimed at exploiting the influence of author writing patterns in book recommender system context in terms of author Stylometry. Finally, all the individual modules are judiciously combined to improvise the book recommendation system. The high-level architecture of the proposed framework is depicted in Figure-1.

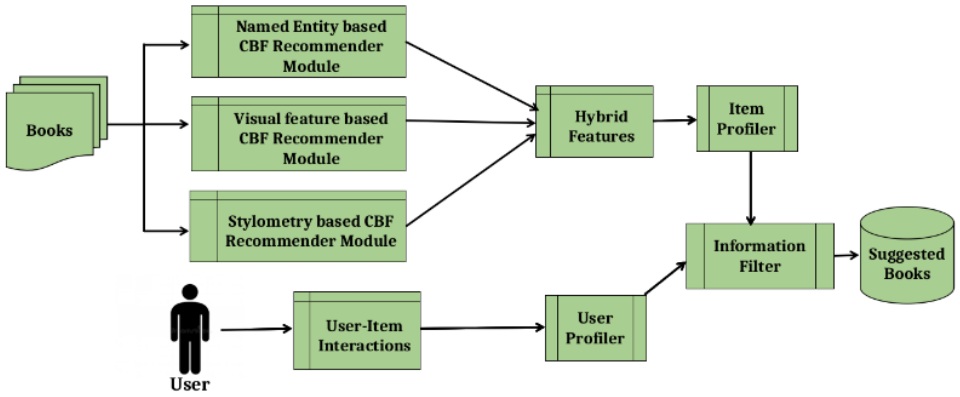

Fig.1: The overall architecture of the proposed model

The methodology behind each module of the proposed framework and its significance in improving the recommendation process is explained in the next sections.

3.1 NER Module

The key idea or hypothesis behind proposing this pipeline is that Named Entities are the key entities of any book that can provide an insight into the reader's possible choices of reading other related books. A named entity generally refers to an entity with a specific meaning or strong reference in the text usually including the organizations, locations, times, date, currency, and percentage. The named entity recognition (NER) system extracts the above entities from the unstructured input text and can identify more of the categories of entities according to the business needs such as product name, company name, price, and so on. NER task is a typical sequence labeling task and the dominant approach of handling sequence labeling task is Conditional Random Fields (CRF). However, a traditional CRF model relies heavily on hand-crafted features, which is time-consuming and hard to develop. Some of the recent works [25] have proposed deep learning techniques such as BiLSTM-CNN, LSTM-CRF models. They have used word-level and character-level word embedding representations along with additional information (e.g., gazetteers, lexical similarity) before feeding into context encoding layers. But these methodologies are time-consuming and resource-intensive. Recently, spaCy Named Entity Recognition system [26] proposed a sophisticated word embedding strategy using sub-word features and "Bloom" embeddings, a deep convolutional neural network with residual connections, and a novel transition-based approach to named entity recognition. Hence, the present work [27] exploits the efficient model proposed by spaCy to extract the named entities from the full-textual content. Another advantage of using spacy is the availability of co-reference resolver as a pipeline. The working mechanism of the NER Module is depicted in Figure-2.

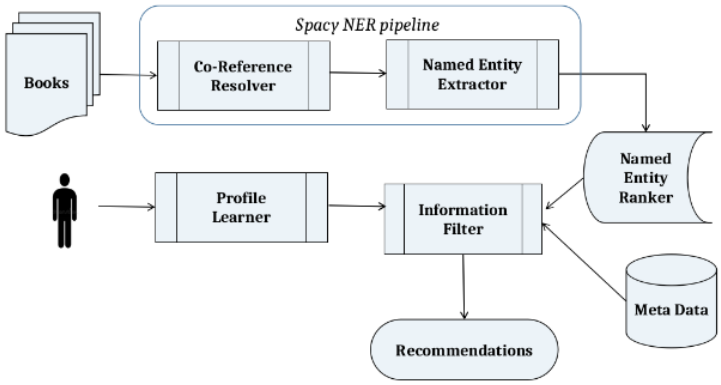

Fig. 2: The working mechanism of NER Module

Books are supplied in plain text format to the co-reference resolver to replace the original entities with their referents. After this process, the modified text is supplied to Spacy Named Entity Extractor to figure out the 18 different categories of named entities present in the book text. The obtained entities are ranked based on their occurrence as well as lexical dispersion. The idea of considering lexical dispersion is to give proper weightage to the entities which are covered throughout the book text. Those ranked entities are utilized for the book recommendation process along with the other metadata features.

3.2 Visual Feature Extraction Module

Even though the textual content in any book plays a vital role in conveying the knowledge wrapped up in that book, the book front cover is the book's first communication to the user. Hence, the visual feature extraction module tries to extricate the visual as well as textual clues that can play a vital role in the recommendation process. Multiple studies [28] have shown that readers are strongly intrigued by front covers that are visually appealing or eye-catching. The cover art of a book can significantly affect the book's sales and readership. There was a popular poll in goodreads.com with 297363 votes for the following question with four different answers.

Question: Would you read a book only based on the cover?

- *Yes. I like to take chances - 116066 votes (39.0%)*
- *Maybe - 140069 votes (47.1%)*
- *Probably not - 35170 votes (11.8%)*
- *Absolutely not - 6058 votes (2.0%)*

Around 86% of people are ready to take a chance to read the book based on its front cover. All these studies and statistics motivated the proposed work to exploit the book front cover in a Content-Based recommendation setup. The proposed model takes book front cover images as input and tries to learn various latent visual features through a variant of CNN and observable object features as well as features from the text description of the image by using deep learning methods. Three feature extractors namely I) Visual feature extractor, II) Object description extractor, and III) OCR-based text extractor is leveraged to generate the book recommendation. The working mechanism of the visual feature extraction module is rendered in Figure-3.

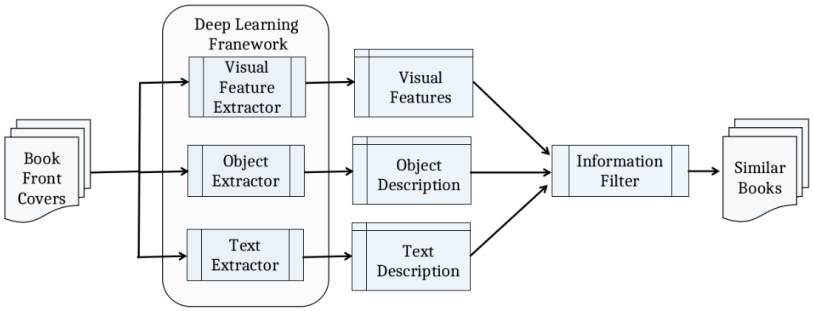

Fig. 3: The working mechanism of Visual Feature Module

The proposed work used a VGG-16 pre-trained model for visual feature extraction from the book front cover image. The obtained visual features are exploited for finding similar books. Then, the proposed model augments with a text description of the front cover image extracted using Faster R-CNN and text appearing on the book front cover using Microsoft Azure OCR module. Microsoft Azure [29] provides APIs for image description and OCR. Finally, all these features are conjointly used in generating book recommendations with the help of the cosine similarity metric.

3.3 Stylometry Module

In general, readers tend to like the same author or similar author books due to a strong interest in the author's writing styles. There are numerous surveys stating that the prominent book selection criterion is based on similar authors or the same authors (read-alike). By considering one of the common factors for book selection, the current work proposes a book recommender system capable of deriving effective recommendations based on the same or similar authors. Even if you have an author list ready in hand, we can't simply generate book recommendations based on the author's names. Since the same author might have written different books belonging to different genres. Along with the author's name, exploring the author's writing style is essential in generating effective book recommendations. Stylometry is one such method to figure out the author's writing style by discovering the regularities that exists in the text. Stylometric research primarily relies on the hypothesis that each individual has a "stylistic fingerprint" that can be quantized and learned. The uniqueness in language may arise clearly because each person's learning and writing capabilities are different from others. The proposed methodology augments the Stylometry features used in Alharthi et al. [30] along with the other extrinsic features proposed by Maity et al. [31]. The overall architecture of the Stylometry model is shown in Figure 4.

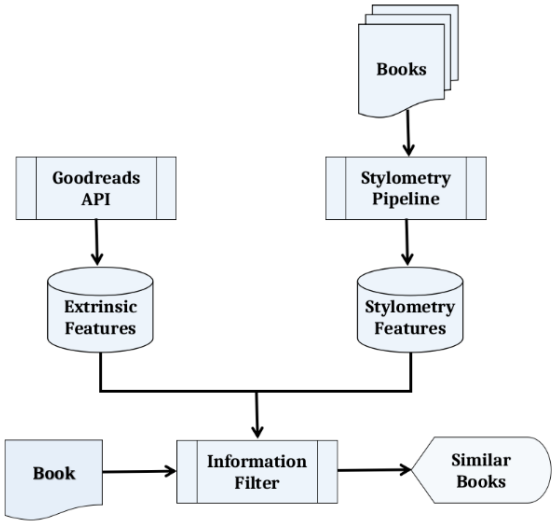

Fig. 4: The working mechanism of Stylometry Module

Initially, various books in plain text format from the LitRec dataset are fed into the Stylometry pipeline to extract various distinguishable features for author identification. In addition to that, the proposed model includes features like user engagement information and author details from Goodreads API to boost the recommendation process. Finally, our dataset comprises 130 features. Some of the Stylometry features that have been used in the proposed work are shown in Table-2.

Table 2: Sample Stylometry Features

Average Word Length	Number of words per paragraph
Average Sentence Length	Number of Exclamations
Lexical Density (type/ token ratio)	Number of paragraphs
Number of dialogs (number of double quote pairs)	Author Gender
Readability Score	Number of Author Awards
% of adjectives, adverbs, nouns, verbs and pronouns	Author Nationality
% of comparatives	Average rating of an author
% of interrogatives	Number of 5 stars
% of unique words	Number of 4 stars
% of stopwords	Number of reviews received by author
% of punctuations	Number of best seller books of the author
% of numerals	Follower count of author
% of Uppercase words (Abbreviations)	User rating entropy of the book
% of non-English words % of quotations	Number of users tagged "currently reading"
% determiners	Number of users tagged "to read"

Finally, all these features are conjointly used in generating book recommendations with the help of the cosine similarity metric.

3.4 Aggrandized Framework

The aforementioned results of Named Entity based CBF Recommender, Visual Feature based CBF Recommender and Stylometry based CBF Recommender are considerable in terms of various metrics like MAE, RMSE, MAP, MAR and Mean F1-score. Later the proposed work tried to explore the effectiveness of the combined recommender by aggregating all the modules. The features extracted using individual recommender modules are concatenated (4096 image features, 18 NER Features, and 130 Stylometry features) into a single vector to build a strong recommendation framework. All the features are normalized to a standard scale in the range [0, 1]. Then, by giving equal weightage to all features, a concatenated feature vector is formed and then recommendations are generated using the K Nearest Neighbor algorithm (K is equal to 10) with Cosine Similarity Metric. Figure-5 shows the concatenation of individual features for generating strong recommendations.

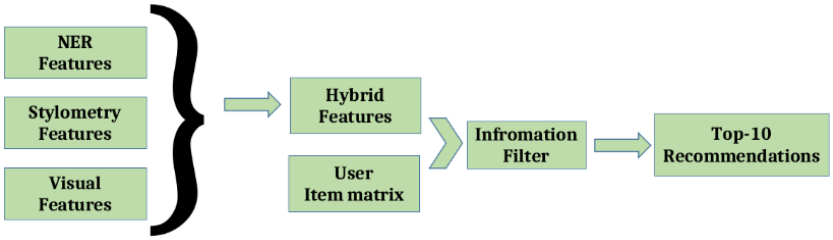

Fig. 5: Concatenation of Various Features to enhance Book Recommendation

4.0 EXPERIMENTATION AND RESULTS

4.1 Dataset used in the present work

The proposed work has considered the LitRec dataset. LitRec dataset [32] is fit for both CF and CBF recommender models that contain a collection of POS tagged literary text of 2598 novels along with their user ratings. These text files in the dataset are curated from *gutenberg.org* and user ratings are extracted from *goodreads.com* for five years.

The detailed description of the dataset is shown in Table-3.

Table 3: Description of the LitRec Dataset

Books	2,598
Users	1927
Ratings	16,042
Sparsity	0.993
Ratings / User	17.01
Ratings / book	6.17

The user-item interactions file consists of details like book read date, book-id (Gutenberg), user-id (Goodreads), rating given by the user, user location, title of the book, file name specified in the text files repository and review posted date. The proposed work refined the dataset at different levels of granularity by eliminating unnecessary as well as redundant information and then augmenting the required information. Each module of the proposed framework needs different kinds of pre-processing steps. Each of those pre-processing steps is discussed in the respective modules.

4.2 Baseline Models for Comparison

The original user-item interaction dataset contains 35484 observations with eight attributes and it has been observed that the majority of ratings are filled with zeros. Since zero ratings don't convey any user preference, the observations having zero ratings have been discarded. After eliminating zero-rating observations, the dataset is left with 17509 observations. The rating distribution of the LitRec dataset is shown in Figure-6.

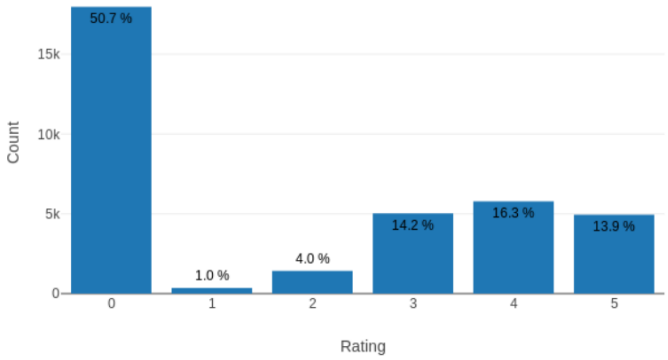

Fig. 6: LitRec dataset Ratings Distribution

To create a CF model, we filtered out the users who have given ratings less than 10 in accordance with the recommender works in the literature [30]. After this step, 14633 observations were left out. On this data, we implemented baseline Collaborative Filtering models with the help of a popular python package called Surprise [33]. To create a CBF baseline model we utilized the metadata extracted from goodreads.com. To generate the recommendations using the baseline CBF model, cosine similarity and K-NN based information filter are used.

4.3 Evaluation Metrics Used for Comparison:

In the proposed work, statistical metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MEA) are used for evaluating the recommendations obtained from the NER Module. Precision, Recall and F-measure are considered for measuring the effectiveness of the recommendations obtained from the Visual feature extraction Module. Accuracy measure is considered for measuring the effectiveness of the recommendations obtained from the Stylometry Module. The Recall is used for evaluating the final recommendations of the proposed system. The reason for considering only recall is to measure the number of recommended books retrieved from the overall useful recommendations which are already preferred by the user.

4.4 NER Module Results

The Named Entity Recognition pipeline is designed to exploit the full-text content of the book by means of Named Entities to get better recommendations. The text files in the existing dataset contain the headers and footers like copyright forms of authors and also parts of speech (POS) tagged. Hence, the input text files are curated by designing a web scraper for the www.gutenberg.org site to extract the required content in plain text format without any irrelevant information. Initially, all the books in plain text format are processed through a spaCy pipeline. spaCy has a state of art NER system which is faster and also allows adding arbitrary classes to the NER pipeline. The novelty of the proposed NER model is employing a co-reference resolver before the Named Entity extractor. The co-reference resolver identifies the expressions in the text that belongs to the same referent. Hence, this step helps in giving proper weightage for all the named entities that appeared in the text. Using the proper weighting mechanism, a candidate named entities are distilled from the Named Entity Ranker. Here, the role of the Named Entity ranker is to extract Named Entities from each category with respect to their frequency as well as lexical dispersion. Lexical dispersion is especially useful for the document collections collected over a longer time period and helps in analyzing how specific terms were used more or less frequently over time. The various categories of the Named Entity tags considered in the proposed model are shown in Table-4 along with a description of each tag.

Table 4: Description of the Named Entity Tags

Entity Tag	**Entity Description**
PERSON	People, including fictional.
NORP	Nationalities or religious or political groups.
FAC	Buildings, airports, highways, bridges, etc.
ORG	Companies, agencies, institutions, etc.
GPE	Countries, cities, states.
LOC	Non-GPE locations, mountain ranges, bodies of water.
PRODUCT	Objects, vehicles, foods, etc. (Not services.)
EVENT	Named hurricanes, battles, wars, sports events, etc.
WORK_OF_ART	Titles of books, songs, etc.
LAW	Named documents made into laws.
LANGUAGE	Any named language.
DATE	Absolute or relative dates or periods.
TIME	Times smaller than a day.
PERCENT	Percentage, including ”%“.
MONEY	Monetary values, including unit.
QUANTITY	Measurements, as of weight or distance.
ORDINAL	“first”, “second”, etc.
CARDINAL	Numerals that do not fall under another type.

For generating the recommendations using the NER module, the named entity information of each book is augmented to the metadata used in the baseline model. The proposed NER-based recommender model has shown significant results when compared to the baseline CF models in the literature, and the results have been shown in Table-5. The proposed NER-based method showed better RMSE and MAE values of 0.87 and 0.69 respectively when compared to the popular SVD-based CF model having RMSE of 0.88 and 0.70. Even this small variation in the performance of the proposed model with a popular collaborative SVD-based approach is appreciable because of the computationally intensive nature of SVD-based approaches.

Table 5: Comparison of Proposed model with respect to other popular ML models

Algorithm	Root Mean Square Error	Mean absolute Error
Proposed Model	*0.876488*	*0.691790*
Singular Value Decomposition (SVD)	0.886648	0.709529
Baseline Only	0.904332	0.733904
Coclustering	0.959614	0.740277
KNN with Means	0.977365	0.736084
KNN with Z-score	0.987120	0.734987
KNN Baseline	0.996418	0.771829
NMF	1.046270	0.829765
KNN Basic	1.073905	0.829069
SlopeOne	1.097981	0.823804
Normal Predictor	1.425743	1.145225

The results obtained from the NER module have also been compared with popularity-based models, pure CF and CBF models in terms of Mean Average Recall@5 and Mean Average Recall@10. The obtained results are shown in Figure-7. It can be observed that the proposed approach surpasses both CF and CBF models. Hence, it can be concluded that the proposed NER model can identify the significant semantic information that can help in book recommendation setup.

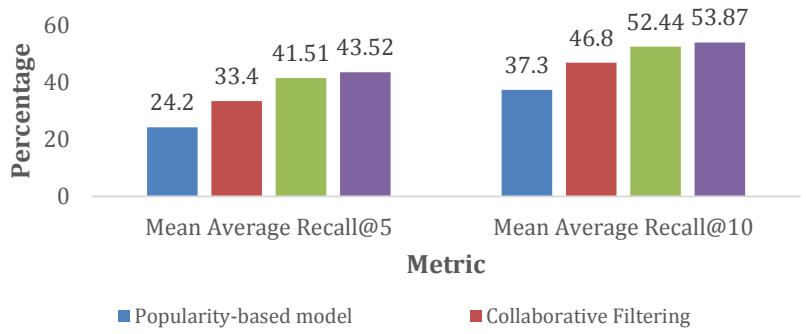

Fig. 7: Comparison of Proposed Model in terms of Mean Average Recall@5 and Mean Average Recall@10

4.5 Visual Feature Extraction Module Results

In the proposed work, we want to exploit the book front cover as one of the features for the recommendation process. The basic idea behind considering the book front cover is that visual features can implicate the book genre. Also, the visual similarity of book covers can provide clues for the recommendation process. The original LitRec dataset does not contain any book front cover images. Hence, the present module scraped the book cover images using Goodreads API. Some of the images that are not available on Goodreads.com have been downloaded manually from amazon.com. Also, since the books in LitRec dataset only belong to the biography and classical genre, the images of top-100 books of 5 more genres like business, cookery, mythology, children and horror from Goodreads.com have been augmented to the original dataset. After this augmentation, the resultant dataset contains 6 different genres named Biography, Business, Cookery, Children, Horror and Mythology. To identify the visual similarity of the book covers, the proposed method used the features extracted using the VGG-16 deep learning framework. The default input size for VGG-16 is 224 X 224 pixels with 3 channels but the current dataset consists of images with different dimensions. For this reason, all the images in the considered dataset are rescaled to the default input size of 224 X 224. The recommendations obtained using visual feature extractor are shown using evaluation metrics such as Mean Average Precision@10 (MAP@10), Mean Average Recall@10 (MAR@10) and Mean Average F-Measure@10 (MAF@10). Figure-8 details about results obtained from visual feature extractor-based recommender.

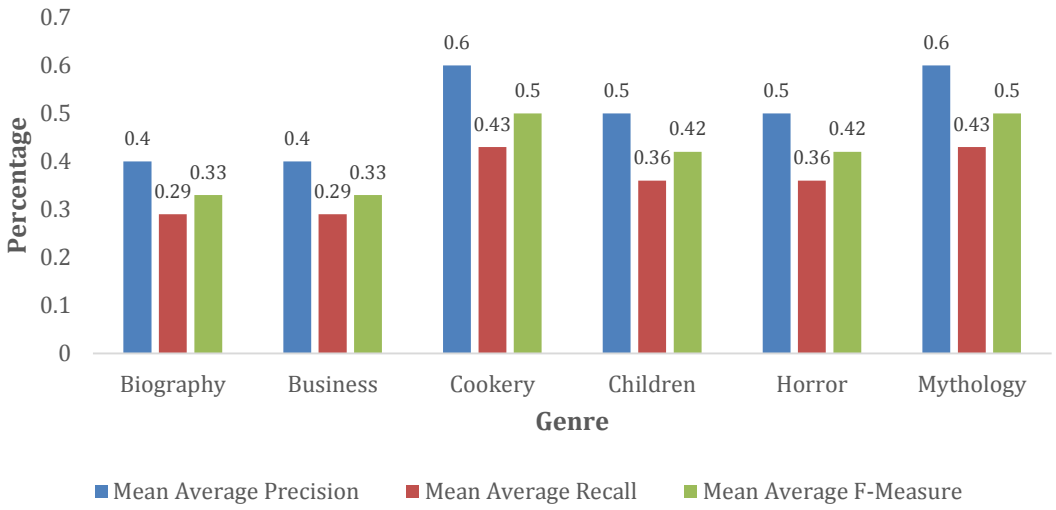

Fig. 8: Mean Average of Precision, Recall and F-Measure before Augmentation

Later, an image description of the book cover is obtained using Microsoft Cognitive Services API. Lastly, using Microsoft OCR API, the text in each image is extracted. Recommendations are generated with the help of all the features extracted from the three methods and compared with the same evaluation metrics. Figure-9 details the results obtained after augmenting the object and OCR information along with VGG-16 features.

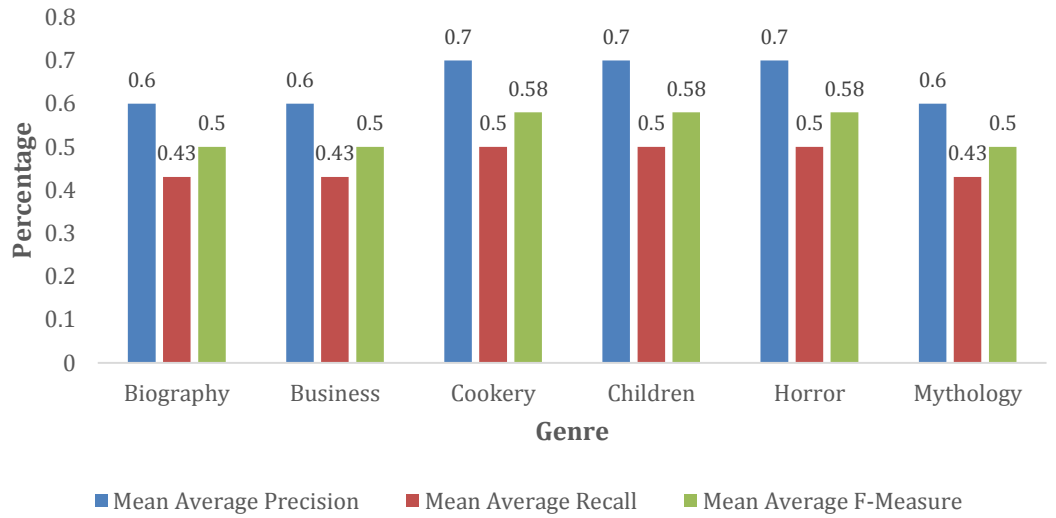

Fig. 9: Mean Average of Precision, Recall and F-Measure after Augmentation

Finally, we [34] compared the results obtained before and after augmenting the object and text descriptions and observed that there is a significant amount of change in the Mean Average Precision, Recall and F-measure for most of the genres. Figure-10 clearly shows that there is an observable change in the F-measure of all the genres except Mythology. From the manual introspection, we identified that there is an overlap in between the Children and the Mythology genre. From the analysis of results, we can conclude that there is a hidden relationship that exists in between the book front cover and its genre which has been established using deep learning frameworks.

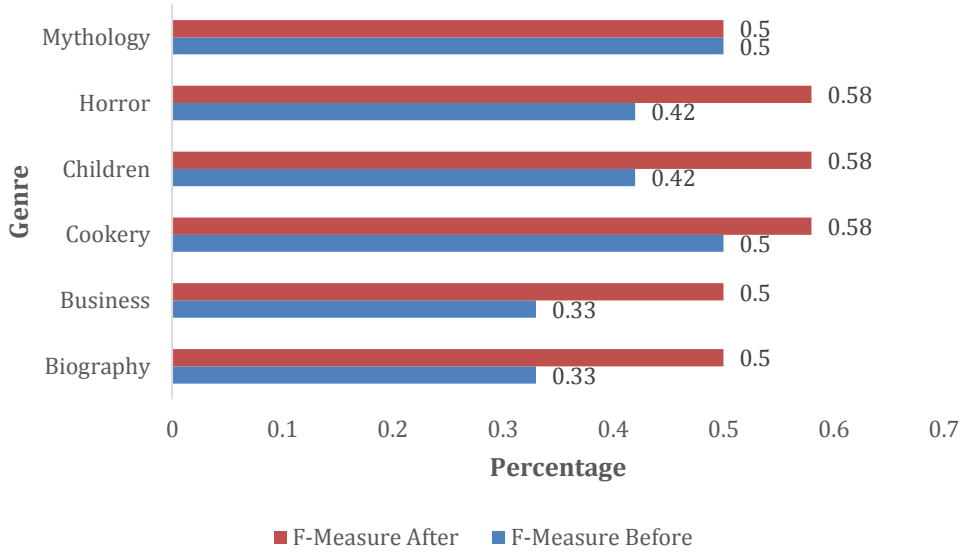

Fig. 10: Change in F-Measure before and after Augmentation of Object and Text Descriptions

4.6 Stylometry Feature Extraction Module

To exploit the Stylometry of authors, the present work considered the LitRec dataset. From the dataset, the users who have rated more than 10 books have been selected and then selected the books whose author has written more than three. Finally, the dataset is having 1050 unique books by 410 different authors rated by 367 distinct users. The proposed model augments the extrinsic features like author information and user engagement details to boost the recommendation process. The basic idea of utilizing these features is to exploit the author's writing style in the book recommendation setup. To verify this, the author's information has been removed and then tried to predict the author_id using various supervised machine learning models like XGB Classifier, K-NN Classifier, Decision Tree Classifier, Linear Discriminant Analysis, Gaussian NB and Random-Forest Classifier. The overall dataset is fed into all the aforementioned classifiers to carry out K-fold (k=10) cross-validation. Initially, we selected 39 candidate features that are positively co-related (greater than 10%) with author_id and performed the classification task. Later, we employed all the 130 features for the classification task and the results are compared using accuracy measure. Figure-11 depicts the comparison of various classifiers with all the features and candidate features. From the results, it can be observed that all the features are collectively playing a pivotal role in author identification than the selected features using correlation. In addition to that, the features identified by us are significant in producing better results than the state-of-art results in the literature [23].

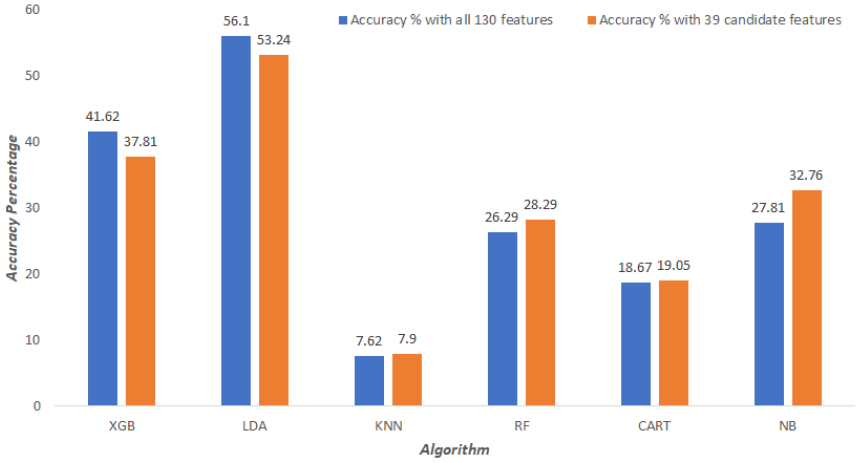

Fig. 11: Accuracy comparison of various algorithms for author_id prediction

The Whisker plot in Figure-12 shows the standard deviation in accuracy for 10-fold cross-validation. Surprisingly, Latent Dirichlet Allocation is able to perform better than all the other models since it is successful in identifying latent features in text useful for author identification.

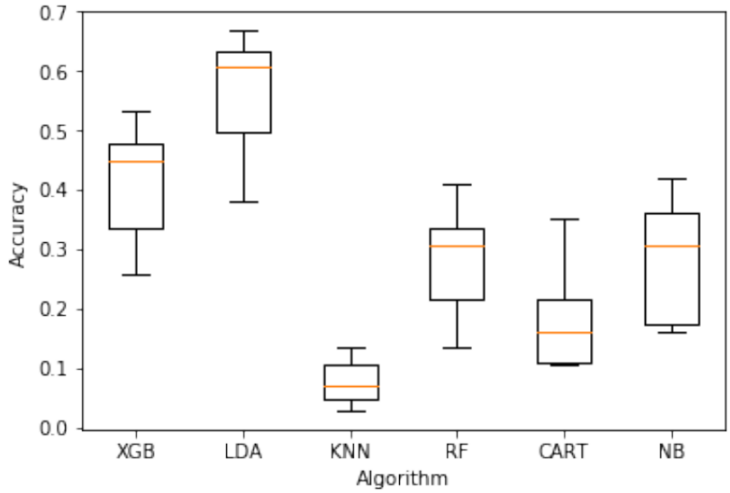

Fig. 12: Accuracy comparison of various algorithms using Whisker plots

Later, the top-10 recommendations given by the Stylometry model have been analyzed to understand how many recommended books are from the same or similar authors. The results are shown in Table-6.

Table 6: Sample recommendations for Author Henry

Input Book and Author	**Top-10 authors and books**		**No. of books from Same Author**	**No. of books from Similar Author**
Allan and the Holy Flower by Henry Rider Haggard	Henry Rider Haggard	A Yellow God: an Idol of Africa	6	4
	Henry Rider Haggard	Montezuma's Daughter		
	Henry Rider Haggard	The Ancient Allan		
	Henry Rider Haggard	Queen Sheba's Ring		
	Henry Rider Haggard	Swallow: a tale of the great trek		
	Henry Rider Haggard	Ayesha: The Return of She		
	Thomas Hardy	The Return of the Native		
	Charles Dickens	Mugby Junction		
	Conan Doyle	Uncle Bernac		
	Anna Katharine Green	Agatha Webb		

A similar author projection is also cross verified with the Literature Map as shown in Figure-14. The Literature-Map is part of the Global Network of Discovery (Gnod) literature recommendation system. The more people like an author and another author, the closer together these two authors will move on to the Literature-Map. It creates a cloud of recommendations in which the names of similar authors will float around the page. The closer they are in style to the chosen writer, the closer they'll be to the middle.

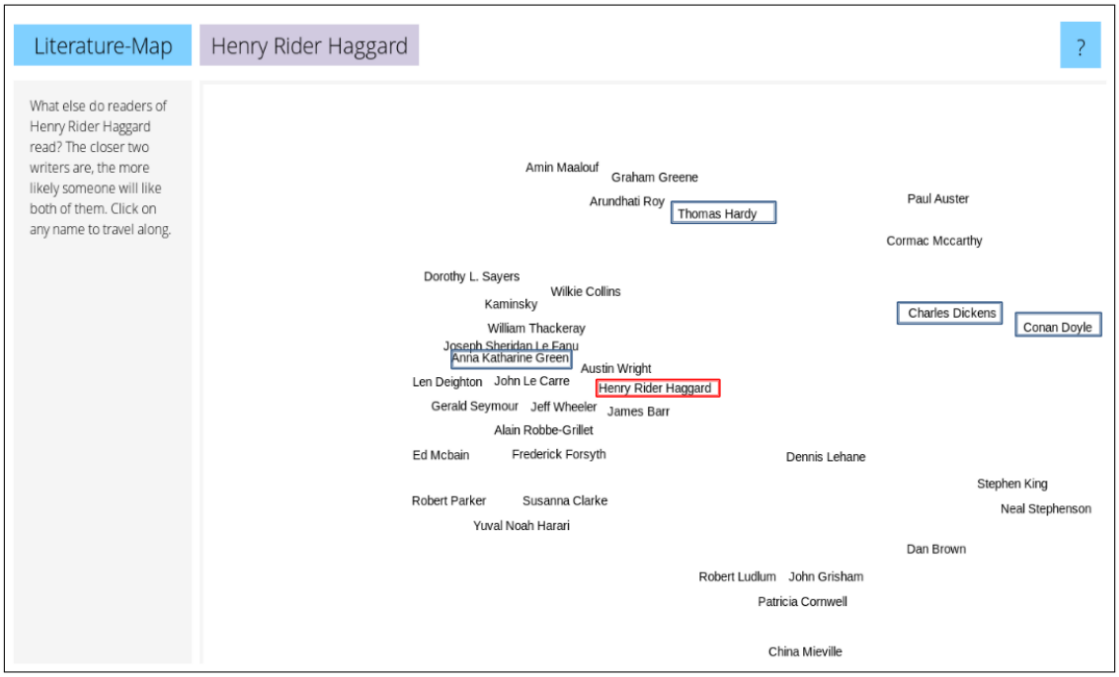

Fig. 13: Similar Authors Projection for Henry Rider Haggard on Literature Map

4.7 Comparison of Aggrandized Framework Results with Baseline and State-of-Art Models

Due to the popularity and efficacy of the deep learning models, the proposed work compared with state-of-art deep learning models in this section. Recently, in the year 2021, Neural Collaborative Filtering with Content Embeddings (NCFCE) [35] and DeepFM [36] filtering models applied on Amazon and KDD2012 Cup dataset to enhance the recommendation process. Due to dataset limitations in mitigating the user behavior, the proposed work is only compared with NCFCE filtering model on LitRec Dataset.

4.7.1 Neural Collaborative Filtering with Content Embeddings (NCFCE)

Neural Collaborative Filtering (NCF) models user-item feature interaction through neural network architecture. The advantage of this model is that NCF can deal with the high level of nonlinearities by adding hidden layers on top of concatenated user-item vectors while learning user-item interactions. Due to multiple hidden layers, the model has sufficient complexity to learn user-item interactions when compared to matrix-factorization based methods. NCF models still suffer with cold-start problem. To handle this limitation content embeddings can be introduced by hybridizing the NCF with content embeddings.

It can be observed from the aforementioned results that each module is able to improve the recommendation process considerably in terms of various metrics like MAE, RMSE, MAP, MAR and Mean F1-score. Finally, the present work tried to observe the effectiveness of the framework by aggregating features from all the modules. The results of the proposed framework are compared with other baseline recommender models as well as each module along with the NCF model and NCF with Content embeddings. It can be observed from the result analysis that the proposed framework significantly improved the Mean Average Recall@10 and the results are very promising. The corresponding results are shown in Figure-14. The experimentation on the LitRec Dataset demonstrates that the proposed aggrandized framework can achieve better recommendations against state-of-the-art algorithms like NCF with Content Embeddings.

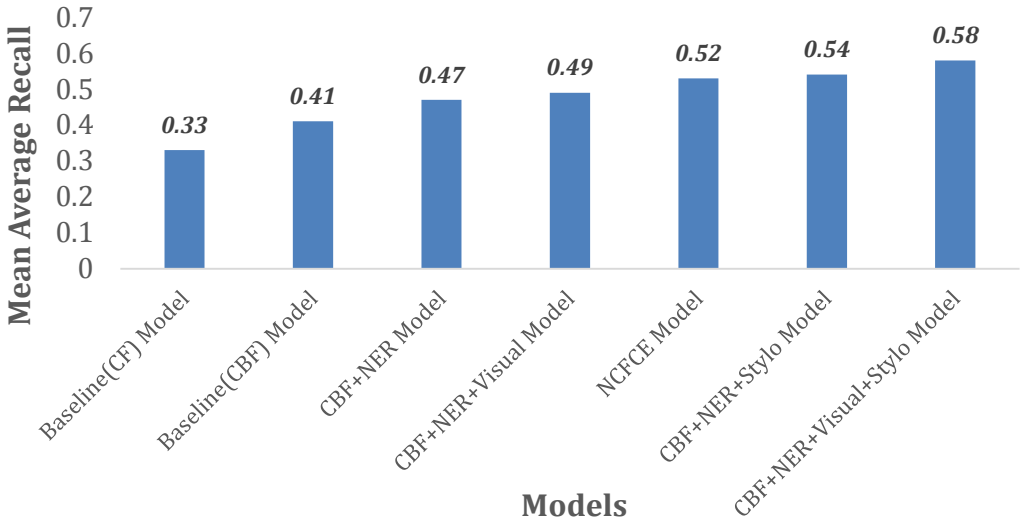

Fig. 14: Comparison of Proposed Method with Baseline Recommenders

It can be observed from the results that augmentation of each module to the CBF model has improved the recommendation results. In the case of NER module, the base line is improved by 6% and in the case of NER combined with Visual Feature extraction module, the baseline is improved by 8%. Finally, in the case of NER module combined with both Visual Feature extraction module and Stylometry module, one can observe that the aggrandized framework is able to improve the overall recommendation accuracy by 18% which supports our claim that the proposed framework is effective in generating book recommendations than the state-of-art recommender models in the literature. It is evident form the results that, the visual and stylometry features enhanced the recommendation process than baseline and hybrid models.

5.0 CONCLUSION

The intent of the proposed research has been to improve the recommendations generated in book domain by using judicious combination the Natural Language Processing and Deep Learning techniques. By using this judicious combination, the feature space for the current CBFs has been enhanced by augmenting with other pertinent features. The various models proposed in the current framework are efficacious to exploit the overall textual content of the book by means of Named Entity based CBF Recommender Module, the genre of the book by means of Visual Feature based CBF Recommender Module and Author Style by means of Stylometry based CBF Recommender Module. From the results shown in section 4.7, it can be observed that the aggrandized framework is able to improve the overall recommendation accuracy by 18% than the baseline models. Also, it can be observed that the proposed model outperforms by 6% than the state-of-art model such as NCF with Content Embedding Model. Hence, it can be concluded that the proposed framework was able to enhance the feature space effectively for the CBF models in the literature. It is also evident that, the visual and stylometry features enhanced the recommendation process than baseline and hybrid models. The proposed work can be further extended by recommending the right sequence of books from the top-10 recommendations generated (Recommendation Post Processing) as well as it can use the user reviews for better user profiling (extracting the expectations of the user from their reviews).

REFERENCES

[1] Hill, Kelly, and Kathleen Capriotti. Social effects of culture: detailed statistical models. Canada Council for the Arts, 2008.

[2] Howard, Vivian. "The importance of pleasure reading in the lives of young teens: Self-identification, self construction and self-awareness." Journal of Librarianship and Information Science 43.1 (2011): 46-55.

[3] Balakrishnan, Vimala, & Hossein Arabi."HyPeRM: A Hybrid Personality-Aware Recommender for Movie." Malaysian Journal of Computer Science, 31.1 (2018): 48-62.

[4] Announcing Goodreads Personalized Recommendations, https://www.goodreads.com/blog/show/303-announcing-goodreads-personalized-recommendations.

[5] Smith, Brent, and Greg Linden. "Two decades of recommender systems at Amazon. com." IEEE internet computing 21.3 (2017): 12-18.

[6] Vaz, P. C., Ribeiro, R. and de Matos, D.M. (2013), Book recommender prototype based on author's writing style, in 'Proceedings of the 10th Conference on Open Research Areas in Information Retrieval', pp. 227-228.

[7] Cho, Erin, and Meng Han. "AI Powered Book Recommendation System." Proceedings of the 2019 ACM Southeast Conference. 2019.

[8] Ekstrand, M. D., Tian, M., Kazi, M. R. I., Mehrpouyan, H. and Kluver, D. (2018), Exploring author gender in book rating and recommendation, in 'Proceedings of the12th ACM Conference on Recommender Systems', ACM, pp. 242-250.

[9] Pankaj Talekar, R. and Deshmukh, M.P.R., 'Hybrid book recommendation engine'. International Journal of Engineering Research and General Science Volume 3, Issue 2, March-April, 2015.

[10] Anwar, Taushif, and V. Uma. "CD-SPM: Cross-domain book recommendation using sequential pattern mining and rule mining." Journal of King Saud University-Computer and Information Sciences (2019).

[11] Ng, Yiu-Kai. "Recommending books for children based on the collaborative and content-based filtering approaches." International Conference on Computational Science and Its Applications. Springer, Cham, 2016.

[12] Tian, Yonghong, et al. "College library personalized recommendation system based on hybrid recommendation algorithm." Procedia CIRP 83 (2019): 490-494.

[13] Sohail, Shahab Saquib, Jamshed Siddiqui, and Rashid Ali. "Book Recommender System Using Fuzzy Linguistic Quantifiers." Applications of Soft Computing for the Web. Springer, Singapore, 2017. 47-60.

[14] Mounika, Addanki, and S. Saraswathi. (2021) "Design of Book Recommendation System Using Sentiment Analysis." Evolutionary Computing and Mobile Sustainable Networks. Springer, Singapore, 2021. 95-101.

[15] Chahinez, B. and Bellot, P. (2015), 'Information retrieval and graph analysis approaches for book recommendation', The Scientific World Journal2015.

[16] Zhang, P., Gu, H., Gartrell, M., Lu, T., Yang, D., Ding, X. and Gu, N. (2016), 'Group-based latent dirichlet allocation (group-lda): Effective audience detection for books in online social media', Knowledge-Based Systems 105, 134-146.

[17] Ng, Yiu-Kai. "CBRec: a book recommendation system for children using the matrix factorization and content-based filtering approaches." International Journal of Business Intelligence and Data Mining 16.2 (2020): 129149.

[18] Miyamoto, Tatsuya, and Daisuke Kitayama. (2019), "A Book Recommendation Method Based on Paragraph Vector and User's Book Arrangement", International Multi Conference of Engineers and Computer Scientists.

[19] Isinkaye, F., Folajimi, Y. and Ojokoh, B. (2015), 'Recommendation systems: Principles, methods and evaluation', Egyptian Informatics Journal 16(3), 261-273.

[20] Ullah, Irfan, and Shah Khusro. "Social book search: the impact of the social web on book retrieval and recommendation." Multimedia Tools and Applications (2020): 1-50.

[21] Musto, C., Greco, C., Suglia, A. and Semeraro, G. (2016), Ask me any rating: A content-based recommender system based on recurrent neural networks, in 'IIR'.

[22] Wadikar, Dhanashri, et al. (2020), "Book Recommendation Platform using Deep Learning.", International Research Journal of Engineering and Technology.

[23] Alharthi, H. and Inkpen, D. (2019), Study of linguistic features incorporated in a literary book recommender system, in 'Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing', ACM, pp. 10271034.

[24] Alharthi, H., Inkpen, D. and Szpakowicz, S. (2018b), 'A survey of book recommender systems', Journal of Intelligent Information Systems 51(1), 139-160.

[25] Li, Jing, et al. "A survey on deep learning for named entity recognition." IEEE Transactions on Knowledge and Data Engineering (2020).

[26] Explosion, A. (2019b), 'spacy's NER model', URL:https://spacy.io/universe/project/video-spacys-ner-model.

[27] Sariki, T. P. and Kumar, G. B. (2018), 'A book recommendation system based on named entities', Annals of Library and Information Studies (ALIS)65(1), 77-82.

[28] Gallagher, D. P. (2015), 'The look of fiction: A visual analysis of the front covers of the newyork times fiction best sellers.

[29] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." IEEE transactions on pattern analysis and machine intelligence 39.6 (2016): 1137-1149.

[30] Alharthi, Haifa, Diana Inkpen, and Stan Szpakowicz. "Authorship identification for literary book recommendations." Proceedings of the 27th International Conference on Computational Linguistics. 2018.

[31] M Maity, S. K., Kumar, A., Mullick, A., Choudhary, V. and Mukherjee, A. (2018), Under-standing book popularity on goodreads,in 'Proceedings of the 2018 ACM Conference on Supporting Groupwork', ACM, pp. 117-121.

[32] Vaz, P. C., Ribeiro, R. and de Matos, D. M. (2012), Litrec vs. movielens-a comparative study., in 'KDIR', pp. 370-373.

[33] Hug N. (2017), Surprise, a Python library for recommender systems, http://surpriselib.com.

[34] Tulasi Prasad Sariki and G. Bharadwaja Kumar, "Exploiting Visual Content of Book Front Cover to Aggrandize the Content Based Book Recommendation System" - International Journal of Innovative Technology and Exploring Engineering (IJITE), Volume-8, Issue-12, October-2019.

[35] Zhang, Yihao, Zhi Liu, and Chunyan Sang. "Unifying paragraph embeddings and neural collaborative filtering for hybrid recommendation." Applied Soft Computing 106 (2021): 107345.

[36] Xu, Jianqiang, Zhujiao Hu, and Junzhong Zou. "Personalized product recommendation method for analyzing user behavior using DeepFM." Journal of Information Processing Systems 17.2 (2021): 369-384.