Improving Big Data Recommendation System Performance Using NLP Techniques With Multi-attributes

Hoger K. Omar[1,2], Mondher Frikha[1,3], Alaa Khalil Jumaa[4]
[1]ENETCOM, ATISP Research Lab, University of Sfax, Sfax, Tunisia
[2]Department of Computer Science, College of Computer Science and Information Technology, University of Kirkuk, Kirkuk, Iraq
[3]Department of Electronics, National School of Electronics and Telecommunications of Sfax, University of Sfax, Tunisia
[4]Database Technology Department, Technical College of Informatics, Sulaimani Polytechnic University, Sulaimani 46001, Kurdistan Region, Iraq
E-mail: omar.hoger@enetcom.u-sfax.tn

Keywords: natural language processing, book recommendation system, big data, collaborative filtering (CF), alternating least squares (ALS), apache spark

Received: October 4, 2023

Due to the wide availability of big data, institutions and companies are currently concentrating on developing highly effective recommender systems for their users. Traditional recommender systems use standard information such as user, item, and ratings. However, this data may not be sufficient for precise results. To enhance accuracy, it is recommended to include additional information such as textual data in the recommendation system. When dealing with large textual data, employing Natural Language Processing (NLP) techniques is essential for effective data analysis. Therefore, this work proposed a novel big data recommender system that enhances collaborative filtering (CF) results by leveraging NLP techniques and dealing with multiple attributes. The study constructs two big data recommendation system models using a machine learning algorithm. In both models, the Alternating Least Squares (ALS) algorithm within the Apache Spark big data tool has been used. The first model did not incorporate NLP techniques, while the second model considered the novel NLP techniques by taking into account the user's review comments. A dataset of more than 3 million ratings and reviews was gathered from the Amazon website with a size of 3.1 GB. The results showed significant improvement after incorporating the suggested NLP-based techniques with multiple attributes.

Povzetek: Predlagan je nov velikopodatkovni priporočilni sistem, ki izboljša rezultate kolaborativnega filtriranja z uporabo tehnik obdelave naravnega jezika in več atributov.

1 Introduction

In the era of information overload, recommendation systems are crucial in connecting users with the content that is most relevant to them [1]. Hence, experts have proposed different viewpoints for assessing the effectiveness of these systems by considering factors such as variety, efficiency, and confidentiality [2]. Social media platforms usually apply recommendation systems in web-based technologies and mobile devices to facilitate highly interactive communication channels. Social media covers various categories including digital libraries, e-commerce platforms, entertainment hubs, forums, social book markets, social review platforms, social games, and social networks. These platforms enable individuals, communities, and organizations to exchange information, provide ratings, and leave comments [3].

Recently, many machine learning (ML) methods have been used in recommendation systems to improve user experience and satisfaction. By applying Artificial Intelligence (AI) these systems can provide superior recommendations in compared to traditional approaches. As a result, a new age has emerged for recommender systems by offering deeper insights into the connections between users and items [4]. A recommendation system comprises a set of software tools designed to suggest items based on the user's preferences and needs. Different techniques such as Jaccard similarity, Cosine similarity, and Pearson similarity can be employed to calculate the similarity between the content and the user's preferences [5]. The systems select the most suitable and appropriate item by analyzing the interconnection between user-based interactions and item-based interactions [6]. Hence, recommendation systems operate based on three primary components: system users, items to be recommended, and interactions between users and the system. These interactions take various forms, and one of the key elements is the rating. The rating holds significant importance as it represents a user's evaluation of an item that the system may recommend [7].

One of the most widely used technologies for recommendation systems is collaborative filtering (CF) and over the period many metaheuristic algorithms have been developed based on it, and some of them have been successful in giving convincingly positive results [8]. CF models evaluate the likenesses and similarities among different users by analyzing their ratings. Subsequently, these models generate predictions for new recommendations by leveraging the connections established among users [9]. The CF can be categorized as Memory-Based Collaborative Filtering and Model-Based Collaborative Filtering. Within Memory-Based Collaborative Filtering, there are two subdivisions: User-Based Collaborative Filtering and Item-Based Collaborative Filtering [10]. On the other hand, the model-based approach employs advanced techniques such as machine learning algorithms to identify patterns within the dataset and utilize that knowledge to process new data. Additionally, matrix factorization (MF) and other relevant approaches are utilized in this context [11].

In this work, the Alternating Least Squares (ALS) within Apache Spark have been utilized. Because the ALS algorithm provides an effective approach for dimensionality reduction in collaborative filtering. Recently, it has been integrated with the latest big data tool Apache Spark MLlib due to its ability to handle the complex computations required by the ALS algorithm [12]. Due to a large number of missing items in the rating data, conventional matrix decomposition algorithms such as Singular Value Decomposition (SVD) encounter serious data fitting challenges when dealing with data sparsity. In contrast, ALS proves to be a highly effective solution for addressing this problem [13]. In addition, the reason for utilizing the ALS algorithm is that it provides a practical approach to handling implicit data that is often not sparse. Furthermore, ALS is an efficient optimization method that is relatively simple to parallelize [14].

On the other hand, Apache Spark stands as one of the most widely used technologies in big data. It achieves parallelism through data structures such as DataFrames, DataSets, and Resilient Distributed Dataset (RDD) which are automatically partitioned and distributed across node clusters [15]. However, Spark implementation allows for faster memory operations than other frameworks such as Apache Hadoop because it supports in-memory (RAM) computations. A comprehensive comparison of Hadoop and Spark frameworks across various Machine Learning algorithms reveals that Spark performs better than Hadoop in the majority of instances [16]. The machine learning (ML) package inside Spark is particularly interesting and offers a wide range of algorithms such as classification, regression, clustering, and collaborative filtering, the architecture of Apache Spark is illustrated in Figure 1 [17]. The data from users is made up of a large volume of text comments, mainly extracted from Streaming Platforms, E-commerce Websites, Mobile Applications, and social media. This valuable and constructive textual information has become a promising and emerging area of research [18].

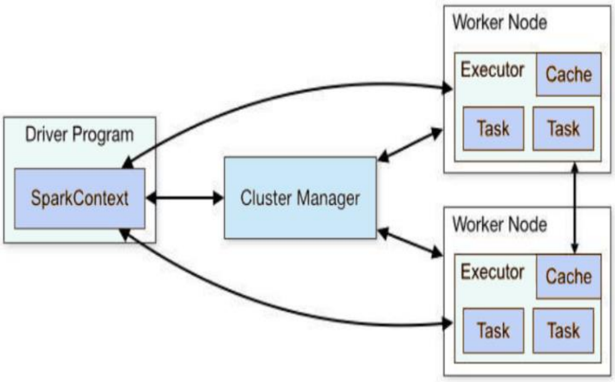


Figure 1: Apache spark architecture.

The traditional recommender systems are developed by building models using standard data, which includes user, item, and user preference information like ratings [19]. The input data mentioned may not be enough to ensure precise results. Relying solely on single-form input data lacks crucial information and is susceptible to noise. Consequently, it is recommended to incorporate additional information into the recommendation system to enhance accuracy [20]. Therefore, this study proposes the utilization of textual data to improve accuracy. When dealing with large volumes of textual data, employing Natural Language Processing (NLP) techniques becomes essential as an initial step for effective data analysis. The significance of NLP in big data processing has been explored and established. NLP employs diverse approaches to interpret the complexities of human language, encompassing techniques like text summarization, sentiment analysis, natural language understanding, and speech recognition [21].

The main contributions of this paper can be summarized as follows:

1- Determine whether incorporating textual feedback genuinely enhances the accuracy of collaborative filtering.
2- Create a natural language processing approach for analyzing textual comments, aiming to enhance the quality of rating prediction in collaborative filtering recommendations.
3- Combine state-of-the-art natural language processing techniques with machine learning to construct an NLP-based recommendation model, and then compare its performance with the traditional model that did not include NLP techniques.

This article is organized as follows: In Section 2, an extensive analysis of the existing literature on big data recommendation systems, with a specific focus on collaborative filtering (CF) methods is presented. Section 3 provides a detailed description of the proposed system architecture. The experimental results are presented in

Section 4. Finally, the conclusion and future work are illustrated in Section 5.

2 Related work

There has been a substantial influx of scientific publications in recent times about big data recommendation systems. These articles have showcased a multitude of strategies for developing impactful systems. The purpose of this section is to examine and appraise the most valued contributions in this particular field.

C. Tegetmeier *et al.* 2023 [22] Examined two AI algorithms for collaborative book recommendation systems, particularly delving into the matrix factorization algorithm utilizing stochastic gradient descent and the book-based k-nearest-neighbor algorithm. They conducted an extensive analysis using the Book-Crossing benchmark dataset, implementing various versions of both algorithms. The goal is to predict unknown book ratings and provide book recommendations to individual users based on the highest anticipated ratings. H. Omar *et al.* 2023 [23] Introduced a cloud-based prototype recommendation system designed to handle big data. Their system employs matrix factorization through three different approaches: singular value decomposition (SVD), alternating least squares (ALS) using Spark, and deep neural network (DNN) utilizing TensorFlow. By tuning the algorithms and parameters in the ALS and DNN approaches they successfully overcome the challenge of dealing with large-scale datasets in collaborative filtering. The outcomes of these two approaches surpass traditional techniques, demonstrating both superior performance and reasonable computational time. P. Sundari *et al.* 2022 [24] Proposed a method to enhance the performance of recommendation systems in the Apache Spark Mllib big data environment. Their approach involves utilizing a hidden behavioral pattern mining technique. By implementing two levels of pre-processing methods, the data size at the item level is significantly reduced. Moreover, the approach effectively tackles the problem of data sparsity in Collaborative Filtering recommendations. N. Osman *et al*. 2021 [25] Utilized a sentiment-based model incorporating contextual information in the Apache Spark platform. To address the issue of domain sensitivity, a new approach was introduced, which involved combining the sentiment-based model with contextual information. An experiment was conducted to compare the effectiveness of the standard rating model, standard sentiment model, and contextual information model. The results indicated that the proposed textual-based model outperformed the traditional collaborative filtering in terms of accuracy. M. Awan *et al*. 2021 [26] Their study implemented a movie recommender system in a big data environment by using CF with the ALS algorithm within Apache Spark. For predicting the top-rated movies, they used a user's latest movie category search data as a reference for training the recommender engine. The approach involved model-based MF and successfully resolves various challenges associated with this method. S. Gosh *et al.* 2020 [27] Introduced an e-commerce recommendation system that applies the ALS algorithm with the MF technique on Apache Spark MLlib. The results prove a substantial decrease in the root mean square error. Consequently, the research shows that the ALS algorithm is well suited for training explicit feedback datasets where users offer ratings for items. Z. JI *et al.* 2019 [28] Proposed the integration of user ratings, reviews, and social data to formulate a hybrid recommendation model in the Spark platform. This model combines Word2Vector for transforming review texts into vectors uses CoDA for community identification, and employs the linear regression (LR) algorithm for rating estimation. Experimental results illustrate that this approach significantly improves the accuracy in compared to a conventional matrix factorization model. K. Dahdouh *et al.* 2019 [29] Designed a distributed course recommender system to aid students in finding appropriate academic resources. Their recommendation system can efficiently manage vast amounts of data and support significant scalability through intensive and massively parallel data processing. They achieve this by leveraging the capabilities of Apache Spark and Apache Hadoop. M. Aljunid *et al*. 2019 [30] Proposed a movie recommender system based on the ALS algorithm with employing Spark. The research explores the optimal selection of the ALS algorithm parameters that can effectively enhance the performance of a robust Recommender System. After analyzing their findings, the study determines how the efficiency of developing the movie recommendation system engine is influenced by the selection of ALS algorithm parameters. Z. Xing *et al.* 2017 [31] Proposed a new approach for generating latent embeddings of podcast content in a content-based recommendation system. This method combines all the text-related features that are associated with the audio and then applies the NLP techniques. By using this approach, the embeddings are created to assess the similarity of content across several podcast items.

On the other hand, this paper focuses on the importance of developing an effective recommender system by institutions and companies that are driven by the accessibility of big data. Traditional systems only depend on standard data such as user, item, and rating information and may fall short in precision. To address this limitation and enhance accuracy, integrating additional data, particularly textual information, becomes crucial. Leveraging NLP techniques is imperative for the effective analysis of large textual datasets. The research introduces a novel big data recommender system that enhances Collaborative Filtering outcomes by integrating NLP techniques with multiple attributes. Two machine learning models were constructed. Both employ the ALS algorithm within Pyspark. The first model did not incorporate NLP techniques, while the second integrated the NLP-based methods by using user review comments. The dataset is comprised of over 3 million ratings and reviews and collected from the Amazon website with a size of 3.1 GB. The results demonstrated significant accuracy improvements when integrating the proposed NLP-based techniques with multiple attributes. This study underscores the importance of utilizing NLP with adding supplementary information, such as review comments to enhance the performance of big data recommender systems.

3 Proposed recommendation system

In this section, the design of an enhanced big data book recommendation system that integrates machine learning with NLP techniques is outlined. The proposed framework includes two forms of ALS algorithms. One with applying NLP techniques and the other without as shown in Figure 2 which explains the overall system steps. In general, the framework encompasses nine distinct steps:

1- The task involves gathering a large dataset, which is approximately 3.1 GB in size and contains over 3 million user ratings from the Amazon website. This dataset is made available to developers and researchers, enabling them to utilize the data for scientific purposes.

2- Typically, the primary dataset may contain numerous unnecessary columns, such as "marketplace" or "verified purchase," which add no value to the work. To accelerate the data processing and computation time, all these irrelevant attributes are removed.

3- To enhance the accuracy, several preprocessing procedures have been applied. These encompass converting the file format from TSV to CSV, conducting data cleaning operations, data normalization, and various other related steps.

4- The objective of this step serves two main purposes: to reduce computational time while maintaining accuracy. To achieve this, only a subset comprising 20% of the large dataset was utilized. This involved dividing the entire dataset into five equal parts, with each part constituting 20% of the whole records, and conducting separate tests on each segment. In total, 10 models were constructed, 5 models utilizing the traditional ALS algorithm, and the other 5 models employing the proposed ALS NLP-based method.

5- In this phase, the required dependencies for machine learning and the Apache Spark platform have been prepared.

6- The work is divided into two distinct methods. The first method utilizes the traditional ALS algorithm, while the second method employs the newly proposed ALS NLP-based approach. In the ALS NLP-based part, various NLP techniques have been applied to the textual comments to enhance the quality of the extracted knowledge. This includes Tokenization, Lemmatization, Stemming, Stop Word Removal, and culminates with Spell Checking, Syntax Checkers, and Word Embeddings.

7- By handling each segment of the data, five lists of recommended books were created for each method. As a result, a total of 10 lists were obtained from both employed methods. Each list corresponds to a specific model.

8- In this phase, each list of recommended books is assessed individually.

9- A comparison was made between the results of the models within each method to evaluate the effectiveness of the NLP-based models in comparison to the models without NLP. This assessment was based on metrics such as Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), the fraction of concordant pairs (FCP), and time performance.

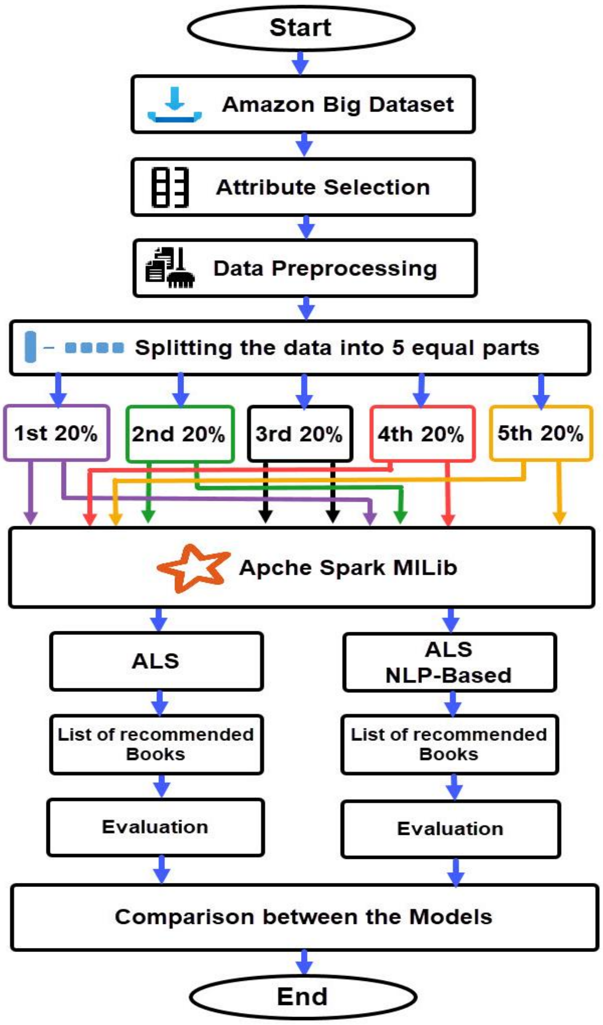


Figure 2: The proposed system architecture and steps.

4 Experimental and results

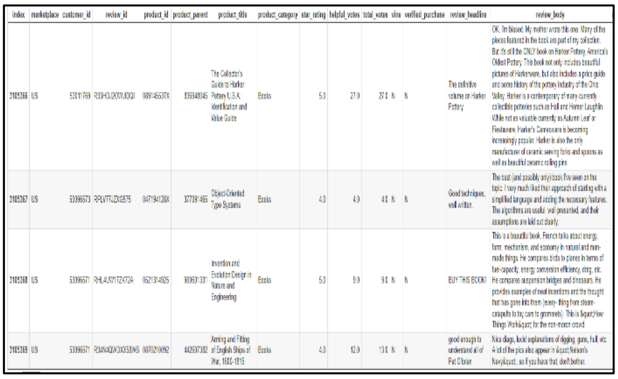
The following subsections, present a comprehensive overview of the experiments, setup, and results. Subsection 4.1 discusses the experimental setup in detail, Subsection 4.2 provides a complete description of the utilized dataset. Finally, subsection 4.3 presents the results and analysis for all the constructed models.

4.1 Experimental setup

Throughout the process of this work, the chosen integrated development environment (IDE) was Google Colab Pro-Plus with a GPU and TPUs runtime environment. This Jupyter Notebook environment comes with a paid version, runs completely in the cloud, and does not require any setup. Through the utilization of Google Colab, users have the ability to write and execute code, save and share their analyses, and directly access powerful computing resources from their web browsers [32]. The platform was specifically designed for Python code development and execution. It offered three subscription plans, comprising a free version and two paid versions: Colab Pro and Colab Pro Plus. The reason for using Colab Pro Plus is due to its provision of additional resources necessary for this project and a higher usage limit when compared to the free version.

4.2 Dataset's description

This study utilized a real dataset obtained from the Amazon website to validate the effectiveness of the proposed methods [33]. The dataset specifically pertains to books and is presented in a Tab-Separated Values File (TSV) format. It contains various attributes, including customer id, product id, product title, star ratings, total votes, review headlines, review bodies, review date, and more. Analyzing this data required employing feature extraction and several natural language processing techniques to achieve satisfactory results. The dataset is quite big, consisting of over 3 million user ratings. The file size is 3.16 GB with 15 attributes and 3,105369 rows. Figure 3 displays a view of the last four rows in the dataset.



index	marketplace	customer_id	review_id	product_id	product_parent	product_title	product_category	star_rating	helpful_votes	total_votes	vine	verified_purchase	review_headline	review_body
3105366	US	53011769	R33HOJ2OWJIDQI	089145537X	836948045	The Collector's Guide to Harker Pottery U.S.A.: Identification and Value Guide	Books	5.0	27.0	27.0	N	N	The definitive volume on Harker Pottery	OK, I'm biased. My mother wrote this one. Many of the pieces featured in the book are part of my collection. But it's still the ONLY book on Harker Pottery, America's Oldest Pottery. This book not only includes beautiful pictures of Harkerware, but also includes a price guide and some history of the pottery industry of the Ohio Valley. Harker is a contemporary of many currently collectible potteries such as Hall and Homer Laughlin. While not as valuable currently as Autumn Leaf or Fiestaware, Harker's Cameoware is becoming increasingly popular. Harker is also the only manufacturer of ceramic serving forks and spoons as well as beautiful ceramic rolling pins
3105367	US	53096573	RPLV77JZXG575	047194128X	377091465	Object-Oriented Type Systems	Books	4.0	4.0	4.0	N	N	Good techniques, well written.	The best (and possibly only) book I've seen on the topic. I very much liked their approach of starting with a simplified language and adding the necessary features. The algorithms are useful, well presented, and their assumptions are laid out clearly.
3105368	US	53096571	RHL4UWY1TZ4T2A	0521314925	980601331	Invention and Evolution Design in Nature and Engineering	Books	5.0	9.0	9.0	N	N	BUY THIS BOOK!	This is a beautiful book. French talks about energy, form, mechanism, and economy in natural and man-made things. He compares birds to planes in terms of fuel-capacity, energy conversion efficiency, drag, etc. He compares suspension bridges and dinosaurs. He provides examples of neat inventions and the thought that has gone into them (every- thing from steam-catapults to toy cars to grommets). This is &quot;How Things Work&quot; for the non-moron crowd.
3105369	US	53096571	R34N4QWDAX53WB	0870210092	442607382	Arming and Fitting of English Ships of War, 1600-1815	Books	4.0	12.0	13.0	N	N	good enough to understand all of Pat O'brien	Nice diags, lucid explanations of rigging, guns, hull, etc. A lot of the pics also appear in &quot;Nelson's Navy&quot;, so if you have that, don't bother.

Figure 3: A screenshot of the last four rows in the dataset.

4.3 Results

The proposed recommendation system was built using matrix factorization and the ALS algorithm, utilizing the Apache Spark machine learning library (MLlib) and the Python programming language. Specifically, the Python implementation is available within the Spark API, known as Pyspark. The driver memory and executor memory of Spark are set to 20 gigabytes. In this work, two ALS models were constructed: the first model did not utilize NLP techniques which means using a traditional technique by taking into account only user-item-rating without the review text. On the other hand, the second model which is the proposed work incorporates NLP techniques that consider the review comments made by the user. This means the ALS NLP-based model utilized user-item-rating triplets as well as text reviews.

It is worth mentioning that the entire dataset was divided into five equal segments, each representing 20% of the total data. For every segment, separate tests were conducted. The purpose of this division was twofold: to reduce computational time while maintaining accuracy levels. Thus, only a 20% subset of the dataset was used for each set of tests. In total, 10 models were constructed, consisting of five models using NLP-based and five without NLP. Notably, the results obtained from these dataset segments showed a high degree of similarity to each other.

Any recommendation system is regression-based, adding metrics such as accuracy is inappropriate. State-of-the-art research in recommendation systems commonly utilizes RMSE and MAE. This work opted to include these metrics along with FCP for a more comprehensive assessment of system reliability. Furthermore, in order to evaluate the time performance of the recommender system in each model, the execution time was calculated. Hence, all these metrics provided valuable insights into how well the recommender system performed.

The results of the ten constructed models as shown in Table 1 demonstrated that the ALS NLP-based models outperformed the ALS models that did not incorporate NLP techniques in the entire five parts. Fortunately, both RMSE and MAE metrics were drastically decreased to be closer to zero and the FCP metric is increased in the proposed NLP-based model as shown in Figure 4. But, the sole disadvantage of the ALS NLP-based is its high computational time for model building as shown in Figure 5.

Nevertheless, it is worth mentioning that the observed increase in computational time during the initial model-building phase can be attributed to the intricate processes involved in constructing a robust and accurate model. Tasks such as feature extraction, and parameter tuning demand substantial computing resources, contributing to the initial time-intensive nature of this phase. However, it's crucial to emphasize that this high computational demand is a one-time investment. Consequently, Once the model is successfully built its deployment and application to new data showcase a substantial reduction in processing times.

Table 1: Comparison between the traditional ALS model and the proposed ALS NLP-based model.

Dataset part	Type	RMSE	MAE	FCP	Time In Min.
1st 20% Records	ALS	0.76	0.58	0.49	7.12
	ALS NLP-based	0.55	0.49	0.54	101.36
2nd 20% Records	ALS	0.77	0.59	0.48	7.09
	ALS NLP-based	0.56	0.49	0.54	101.13
3rd 20% Records	ALS	0.75	0.57	0.49	7.18
	ALS NLP-based	0.54	0.48	0.53	102.53
4th 20% Records	ALS	0.76	0.58	0.49	7.01
	ALS NLP-based	0.54	0.49	0.54	101.39
5th 20% Records	ALS	0.75	0.57	0.50	7.28
	ALS NLP-based	0.54	0.47	0.55	102.58

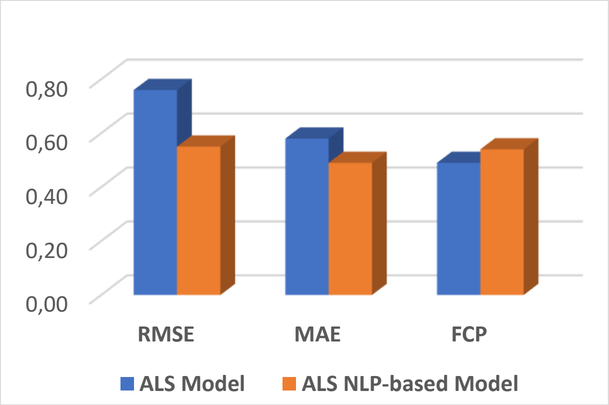


Figure 4: Results of the ALS model and ALS NLP-based model.

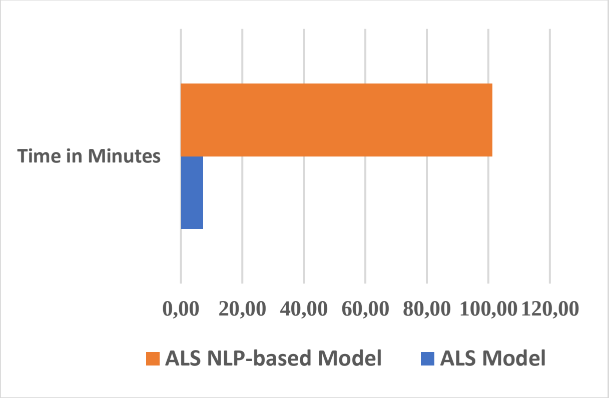


Figure 5: Time performance of ALS model and ALS NLP-based model.

5 Discussion

The existing collaborative recommender system research avoids incorporating textual reviews with other attributes to enhance accuracy. The primary reason is the complexity of the system, especially in Big Data, which demands significant resources. In this system, the utilization of various text preprocessing steps, coupled with natural language processing techniques, has reduced complexity and execution time. Consequently, only 20% of the Big Data is utilized, striking a balance between processing time and accuracy by employing a shrinking step in the preprocessing. Shrinking involves reducing the content of selected attributes during the feature extraction phase while retaining their values. Therefore. the integration of NLP techniques and multi-attributes in Social Big Data recommendation systems holds significant promise for enhancing their performance. NLP empowers these systems to extract meaningful insights from the vast pool of unstructured textual data generated by users. This leads to better contextual understanding, enabling more accurate recommendations.

6 Conclusion

This paper highlights the increasing focus of institutions and companies on developing highly effective recommender systems due to the widespread availability of big data. Traditional recommender systems using standard information such as user, item, and ratings may not always yield precise results. To address this limitation and enhance accuracy, the recommendation system should include additional information such as textual data. Employing Natural Language Processing techniques becomes crucial when dealing with large textual data for effective data analysis. The article presents a novel big data recommendation system that improves collaborative filtering outcomes by using NLP techniques with multiple attributes. Two types of machine learning models were constructed for the recommendation system. Both models applied the Alternating Least Squares (ALS) algorithm within Pyspark. The first type did not incorporate NLP techniques, while the second type considered NLP-based techniques by integrating user review comments. The dataset used for evaluation contained more than 3 million ratings and reviews and was gathered from the Amazon website with a size of 3.1 GB. The results demonstrated significant enhancements in accuracy after integrating the proposed NLP-based techniques with multiple attributes. This research emphasizes the importance of using NLP and supplementary information such as review comments to enhance the performance of big data recommender systems. Future work will concentrate on constructing NLP-based models with deep learning algorithms in TensorFlow and PyTorch frameworks. Furthermore, attention will be directed to using hyperparameter optimization algorithms such as the Optuna optimization algorithm.

References

[1] Deepjyoti Roy, Mala Dutta, "A systematic review and research perspective on recommender systems," *Journal of Big Data*, vol. 9:59, pp. 1-36, 2022, doi.org/10.1186/s40537-022-00592-5.

[2] Yifan Wang, Weizhi Ma, Min Zhang, Yiqun Liu, Shaoping Ma, "A Survey on the Fairness of Recommender Systems," *ACM Transactions on Information Systems,* vol. 41, no. 307, p. 1–43, 2023.

[3] Anitha Anandhan, Liyana Shuib, Maizatul Akmar Ismail, Ghulam Mujtaba, "Social Media Recommender Systems: Review and Open Research Issues," *IEEE Access,* vol. 6, pp. 15608 - 15628,10.1109/ACCESS.2018.2810062, 2018 .

[4] Qian Zhang, Jie Lu, Yaochu Jin, "Artificial intelligence in recommender systems," *Complex & Intelligent Systems,* vol. 7, p. 439–457, 2021, doi.org/10.1007/s40747-020-00212-w.

[5] Sercan Aygün, Savaş Okyay, "Improving the pearson similarity equation for recommender systems by age parameter," in *2015 IEEE 3rd Workshop on Advances in Information, Electronic and Electrical Engineering (AIEEE),* Riga, Latvia, 13-14 November 2015, https://doi.org/10.1109/AIEEE.2015.7367282.

[6] Nur W. Rahayu, Ridi Ferdiana, Sri S. Kusumawardani, "A systematic review of ontology use in E-Learning recommender system," *Computers and Education: Artificial Intelligence,* vol. 3, pp. 1-16, 2022, doi.org/10.1016/j.caeai.2022.100047.

[7] Mario Casillo, Brij B. Gupta, Marco Lombardi, Angelo Lorusso, Carmine Valentino, "Context Aware Recommender Systems: A Novel Approach Based on Matrix Factorization and Contextual Bias," *Electronics,* Vols. 11,1003, pp. 1-19, 2022, doi.org/10.3390/electronics11071003.

[8] Urvish Thakker, Ruhi Patel, Manan Shah, "A comprehensive analysis on movie recommendation system employing collaborative filtering," *Multimedia Tools and Applications,* vol. 80, p. 28647–28672, 2021, doi.org/10.1007/s11042-021-10965-2.

[9] Yongheng Mu, Yun Wu, "Multimodal Movie Recommendation System Using Deep Learning," *Mathematics,* vol. 11, pp. 1-12, February 2023.

[10] Hyeyoung Ko, Suyeon Lee, Yoonseo Park, Anna Choi , "A Survey of Recommendation Systems: Recommendation Models, Techniques, and Application Fields," *electronics,* Vols. 11, 141, pp. 1-48, 2022, doi.org/10.3390/electronics11010141.

[11] Pegah Malekpour Alamdari, Nima Jafari Navimipour, Mehdi Hosseinzadeh, , Ali Asghar Safaei, Aso Darwesh, "A Systematic Study on the Recommender Systems in the E-Commerce," *IEEE Access,* vol. 8, pp. 115694-115716, 2020, 10.1109/ACCESS.2020.3002803.

[12] Y. Niu, "Collaborative Filtering-Based Music Recommendation in Spark Architecture," *Hindawi Mathematical Problems in Engineering,* vol. 2022, pp. 1-8, 2022 doi.org/10.1155/2022/9050872.

[13] Hanmin Ye, Qiuling Zhang, Xue Bai, "A new collaborative filtering algorithm based on modified matrix factorization," in *IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC),* Chongqing, China, 2017, 10.1109/IAEAC.2017.8053995.

[14] Jung-Bin Li, Szu-Yin Lin, Yu-Hsiang Hsu, Ying-Chu Huang,, "An empirical study of alternating least squares collaborative filtering recommendation for Movielens on Apache Hadoop and Spark," *Int. J. Grid and Utility Computing,* vol. 11, no. 5, pp. 674-682, 2020.

[15] Jose´ M. Abuı´n, Nuno Lopes, Luı´s Ferreira, Toma´s F. Pena, Bertil Schmidt, "Big Data in metagenomics: Apache Spark vs MPI," *PLOS ONE,* vol. 15(10), pp. 1-20, 2020, doi.org/10.1371/journal.pone.0239741.

[16] Carlos Fernandez-Basso, M. Dolores Ruiz, Maria J. Martin-Bautista, "Spark solutions for discovering fuzzy association rules in Big Data," *International Journal of Approximate Reasoning,* vol. 137, pp. 94-112, 2021.

[17] Hoger K. Omar, Alaa Khalil Jumaa, "Distributed big data analysis using Spark parallel data processing," *Bulletin of Electrical Engineering and Informatics,* vol. 11, no. 3, pp. 1505-1515, 2022, DOI: 10.11591/eei.v11i3.3187.

[18] Hero O. Ahmad, Shahla U. Umar, "Sentiment Analysis of Financial Textual Data Using Machine Learning and Deep Learning Models," *Informatica,* vol. 47, p. 153–158, 2023, doi.org/10.31449/inf.v47i5.4673.

[19] Yong Zheng, David Wang, "Multi-Criteria Ranking: Next Generation of Multi-Criteria Recommendation Framework," *IEEE Access,* vol. 10, pp. 90715-90725, 2022, 10.1109/ACCESS.2022.3201821.

[20] Zhen Tian, Lamei Pan, Pu Yin, Rui Wang, "Information Fusion-Based Deep Neural Attentive Matrix Factorization Recommendation," *Algorithms,* vol. 14(10), pp. 1-17, 2021, doi.org/10.3390/a14100281.

[21] Sebastião Pais, João Cordeiro, M. Luqman Jamil, "NLP-based platform as a service: a brief review," *Journal of Big Data,* vol. 9, no. 54, pp. 1-26, 2022.

[22] Clemens Tegetmeier, Arne Johannssen, Nataliya Chukhrova, "Artificial Intelligence Algorithms for Collaborative Book Recommender Systems," *Annals of Data Science,* pp. 1-36, 2023, doi.org/10.1007/s40745-023-00474-4.

[23] Hoger K. Omar, Mondher Frikha, Alaa Khalil Jumaa, "Big data cloud-based recommendation system using NLP techniques with machine and deep learning," *Telkomnika Telecommunication Computing Electronics and Control*, vol. 21, no. 5, p. 1076~1083, 2023, DOI: 10.12928/TELKOMNIKA.v21i5.24889.

[24] P. Shanmuga Sundari , M. Subaji, "An improved hidden behavioral pattern mining approach to enhance the performance of recommendation system in a big data environment," *Journal of King Saud University – Computer and Information Sciences,* vol. 34, no. 10, pp. 8390-8400, 2022, doi.org/10.1016/j.jksuci.2020.09.010.

[25] Nurul Aida Osman, Shahrul Azman Mohd Noah, Mohammad Darwich, Masnizah Mohd, "Integrating contextual sentiment analysis in collaborative recommender systems," *PLOS ONE,* vol. 16, no. 3, pp. 1-21, March 2021.

[26] Mazhar Javed Awan et al, "A Recommendation Engine for Predicting Movie Ratings Using a Big Data Approach," *electronics MDPI,* vol. 10, pp. 1-17, 2021, doi.org/10.3390/electronics10101215.

[27] Subasish Gosh, Nazmun Nahar, Mohammad Abdul Wahab, Munmun Biswas, Mohammad Shahadat Hossain, Karl Andersson , "Recommendation System for E-commerce Using Alternating Least Squares (ALS) on Apache Spark," in *International Conference on Intelligent Computing & Optimization ICO 2020,* Koh Samui, Thailand, 2020, doi.org/10.1007/978-3-030-68154-8_75.

[28] Zhenyan Ji, Huaiyu Pi, Wei Wei, Bo Xiong, Marcin Wozniak, Robertas Damasevicius, "Recommendation Based on Review Texts and Social Communities: A Hybrid Model," *IEEE Access,* vol. 7, pp. 40416-40427, 2019.

[29] Karim Dahdouh, Ahmed Dakkak, Lahcen Oughdir, Abdelali Ibriz, "Large-scale e-learning recommender system based on Spark and Hadoop," *Journal of Big Data,* vol. 6:2, pp. 1-23, 2019, doi.org/10.1186/s40537-019-0169-4.

[30] Mohammed Fadhel Aljunid, D. H. Manjaiah, "Movie Recommender System Based on Collaborative Filtering Using Apache Spark," in *Data Management, Analytics and, Innovation. Advances in Intelligent Systems and Computing, vol 839*, Singapore, 2019, doi.org/10.1007/978-981-13-1274-8_22.

[31] Zhou Xing, Marzieh Parandehgheibi, Fei Xiao, Nilesh Kulkarni, Chris Pouliot, "Content-based Recommendation for Podcast Audio-items using Natural Language Processing Techniques," in *2016 IEEE International Conference on Big Data (Big Data)*, Washington, DC, USA, 2017.

[32] Teddy Surya Gunawan, Arselan Ashraf, Bob Subhan Riza, Edy Victor Haryanto, Rika Rosnelly, Mira Kartiwi, Zuriati Janin, "Development of video-based emotion recognition using deep learning with Google Colab," *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, vol. 18, no. 5, p. 2463~2471, October 2020,.

[33] "Amazon," [Online]. Available: https://s3.amazonaws.com/amazon-reviews-pds/tsv/index.txt. [Accessed 8 4 2023].