Recommender System In Youtube Based On Sentiment Based Model

Sariki Ravi Teja
PG scholar, Department of MCA, CDNR collage, Bhimavaram, Andhra Pradesh.
K.Suparna
(Assistant Professor), Master of Computer Applications, DNR collage, Bhimavaram, Andhra Pradesh.

Abstract

In the era of personalized digital experiences, recommender systems play a pivotal role in enhancing user engagement. Traditional recommendation techniques heavily rely on user ratings, which often suffer from limitations such as the cold start problem, leading to inaccurate suggestions when user data is sparse or inconsistent. To address this challenge, this project proposes a sentiment-based model that incorporates user comments alongside ratings to improve recommendation accuracy. By analyzing sentiments expressed in comments, deeper insights into user preferences are obtained, enabling more reliable predictions. A CNN2D (Convolutional Neural Network 2D) model is trained on YouTube comment data categorized into five sentiment levels, from negative to extremely happy, allowing a nuanced interpretation of user feedback. The model's performance is evaluated using RMSE (Root Mean Square Error) metrics, indicating prediction accuracy with dynamically split training and testing datasets. The application, built with Python and MySQL, offers functionalities such as user registration, dataset loading, model training, and both file-based and single-comment analysis for sentiment prediction and movie recommendation. With a user-friendly interface and detailed modules, the system allows seamless interaction, accurate sentiment classification, and effective recommendation generation. This approach not only mitigates the cold start issue but also enriches the recommendation quality by leveraging the emotional context found in user comments.

INTRODUCTION

In today's digital world, recommendation systems have become an essential part of many online platforms, helping users discover products, services, and content based on their preferences and behavior. These systems use various algorithms to suggest relevant items by analyzing user data such as browsing history, ratings, and past interactions. However, traditional recommendation systems that rely mainly on ratings often face significant challenges, especially the cold start problem. This issue arises when there is insufficient user data, leading to poor recommendation accuracy and user dissatisfaction.

One of the major limitations of conventional recommendation systems is the dependency on a single factor—ratings. Ratings alone fail to capture the full depth of user sentiment and opinions about a product or service. Users often leave detailed comments that express their true feelings and experiences, which ratings might not fully reflect. Ignoring this rich source of information results in incomplete analysis and less effective recommendations. To overcome these challenges, there is a growing interest in combining ratings with sentiment analysis of user comments.

In this project, we propose a sentiment-based model that enhances traditional recommendation systems by incorporating both user ratings and comments. Sentiment analysis, which involves examining the emotions expressed in text, provides valuable insights into user preferences. By leveraging comments from YouTube videos, we can predict user sentiments more accurately and recommend content that aligns better with their emotional responses. This approach aims to minimize the cold start problem and improve overall recommendation quality.

To achieve this, we employ an advanced machine learning technique known as Convolutional Neural Networks (CNN2D). CNNs are highly effective for analyzing patterns in text data and are particularly suitable for sentiment classification tasks. We preprocess YouTube comment datasets and categorize the sentiments into five distinct levels: negative, neutral, positive, happy, and extremely happy. The CNN model is trained on this labeled data to learn the relationship between comment text and sentiment labels, enabling it to predict sentiments of new comments efficiently.

The performance of our CNN-based model is evaluated using the Root Mean Square Error (RMSE) metric, which measures the difference between actual and predicted values. A lower RMSE score indicates better prediction accuracy, and by using dynamic training and testing splits, we ensure that the model's performance remains robust across different datasets. In addition, we have developed a user-friendly web application that allows users to sign up, log in, upload datasets, train the CNN model, and predict sentiments for both multiple comments and individual comments in real-time.

Through this work, we demonstrate that incorporating sentiment analysis into recommendation systems significantly enhances their accuracy and reliability. By tapping into the emotional dimension of user feedback, our system provides more personalized and relevant recommendations. This innovative approach represents a meaningful step forward in the evolution of intelligent recommendation technologies, offering better user experiences and deeper engagement across various digital platforms.

LITERATURE SURVEY

1. Pang et al. (2002) introduced one of the earliest machine learning approaches for sentiment classification using text features. Their work used Naïve Bayes, Maximum Entropy, and Support Vector Machines (SVMs) to classify sentiments expressed in movie reviews. Their findings laid the foundation for integrating text analysis into recommendation systems,

highlighting that textual sentiment carries deeper user opinions than numeric ratings alone.

2. Liu (2012) published a comprehensive survey on sentiment analysis and opinion mining, underlining the importance of extracting subjective information from user-generated content. Liu's work emphasized that emotional content such as YouTube comments holds significant value in understanding user preferences, setting a baseline for developing sentiment-driven recommender models.

3. McAuley and Leskovec (2013) presented a model combining ratings and text reviews to improve product recommendations. They proposed the HFT (Hidden Factors as Topics) model that jointly modeled user ratings and review texts, demonstrating that sentiment extracted from text reviews leads to more accurate and personalized recommendations — an idea highly relevant for sentiment-based YouTube recommender systems.

EXISTING METHOD

Traditional recommendation systems primarily rely on collaborative filtering and content-based filtering techniques. Collaborative filtering predicts a user's interest based on the preferences of similar users, while content-based filtering recommends items similar to those a user has liked in the past. Although these techniques have achieved considerable success, they depend heavily on explicit user ratings to generate accurate recommendations. A significant drawback of these systems is the cold start problem, which occurs when there is insufficient data about a new user or item, leading to poor recommendation quality. Furthermore, relying solely on numeric ratings fails to capture the user's emotions or the reasons behind their preferences.

In some advanced models, matrix factorization techniques like Singular Value Decomposition (SVD) are used to extract latent factors from the user-item rating matrix to improve recommendation quality. However, even these methods suffer from sparsity issues when the number of users or items increases significantly, or when user feedback is limited. Additionally, traditional systems overlook the valuable insights hidden in textual feedback such as comments and reviews, which often contain detailed sentiments that ratings alone cannot reveal. As a result, recommendations made purely on numeric values lack depth and fail to truly align with user emotions and satisfaction levels.

Various attempts have been made to incorporate textual data, but many earlier systems treated comments as auxiliary information without deeply analyzing the sentiments expressed within them. Basic keyword matching or simple polarity classification methods were used, which often led to inaccurate sentiment interpretations. Without robust sentiment analysis, these systems could not fully utilize the emotional context provided by users. Therefore, existing methods still fall short when it comes to providing highly personalized, emotionally intelligent recommendations that truly understand and predict user preferences.

PROPOSE METHOD

The proposed method aims to enhance traditional recommender systems by integrating sentiment analysis of user comments along with conventional rating-based predictions. Instead of solely depending on numeric ratings, which often lack emotional depth and can be sparse for new users or items (leading to the cold start problem), this model uses the sentiments expressed in user comments to provide a more accurate and insightful recommendation. By doing so, it not only captures the user's quantitative evaluation but also the qualitative emotions behind their choices, resulting in a richer understanding of user preferences.

To perform sentiment analysis effectively, we employ a Convolutional Neural Network 2D (CNN2D) model. The YouTube comments dataset is used for training and contains both textual comments and corresponding ratings. Before training, the comments are preprocessed and classified into five sentiment categories: 1 for Negative, 2 for Neutral, 3 for Positive, 4 for Happy, and 5 for Extremely Happy. This structured classification allows the CNN model to learn different emotional intensities and contexts from the comment data, making the sentiment prediction more precise and impactful for the recommendation engine.

The CNN model's training involves splitting the available dataset dynamically into training and testing sets. This dynamic division ensures that the model is tested under varying conditions, promoting its ability to generalize well to unseen data. The model's performance is evaluated using the Root Mean Square Error (RMSE) metric, where a lower RMSE score indicates higher accuracy between predicted and actual sentiment values. The trained model can then predict the sentiment of new user comments, which, when combined with ratings, can generate a set of top recommended videos or products matching the user's emotional and rating profile.

The system is implemented through a user-friendly web application where users can sign up, upload datasets, train the CNN model, and input either multiple or single comments for sentiment analysis and recommendation generation. Users can upload a file containing test comments, and the system will predict sentiments and provide a list of top recommended movies or videos accordingly. Alternatively, users can enter a single comment manually to receive immediate sentiment analysis and recommendations. This modular design makes the application flexible, scalable, and easy to use, providing a significant improvement over traditional recommendation methods.

Result

Now-a-days all application will be using some kind of recommendation system to entice their customers with offers based on their past browsing or collaborative filtering. Existing techniques often suffer from Cold Start issue which will give inaccurate recommendation when matrix size goes smaller or higher. This issue occur because of single entity usage called RATINGS

To overcome from above issue author of this paper employing Comments Sentiments along with ratings. Comments often express user sentiments which can help in getting accurate recommendation. Comments help in predicting accurate sentiment which will help in accurate prediction of Recommendation.

To predict sentiments and recommendation we are employing CNN2D (convolution neural networks) advance algorithm which will get trained on YouTube comments and this comments we have divided into 5 different sentiments from 1 to 5 where 1 refers to Negative, 2 refers to Neutral, 3 refers to Positive, 4 refers to happy and 5 refers to extremely happy.

CNN algorithm performance is evaluated in terms of RMSE (root mean square error) which refers to different between original and predicted values so the lower the difference the better is the algorithm. CNN get tested on dynamic split of train and test data so RMSE score always vary for each run.

To train CNN we are using below YouTube comments dataset

In above dataset screen first row contains dataset column names and remaining rows contains dataset values. So by using above dataset will train and test CNN algorithm.

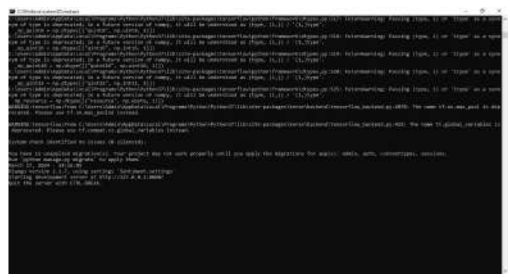
To implement this project we have designed following modules

1) User Sign up: user can sign up with the application
2) User Login: after sign up user can login to application
3) Load Dataset: using this module user can upload and pre-process dataset values
4) Train CNN: using this module user can train CNN algorithm and then get RMSE error as output
5) File Comments Analysis: using this module user can upload test comments file and then CNN will predict sentiments and based on sentiment will predict recommended movies
6) Single comment: user can enter comment text to predict sentiments and movie recommendation

Install python 3.7 and then install all packages given in requirements.txt file and then install MYSQL dataset and then copy content from DB.txt file and paste in MYSQL console to create database

SCREEN SHOTS

To run project double click on run.bat file to get below screen



In above screen server started and now open browser and enter URL as http://127.0.0.1:8000/index.html and press enter key to get below page

In above screen click on 'User Sign up' link to get below page

In above screen user is entering sign up details and then press button to get below page

In above screen user sign up completed and now click on 'User Login' link to get below page

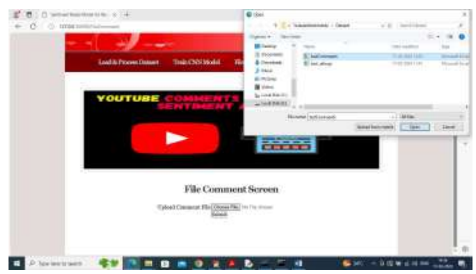
In above screen user is login and after login will get below page

In above screen click on 'Load & Process Dataset' link to get below page

In above screen selecting and uploading 'text_ratings.csv' file and then click on 'Open' button to load dataset and get below page

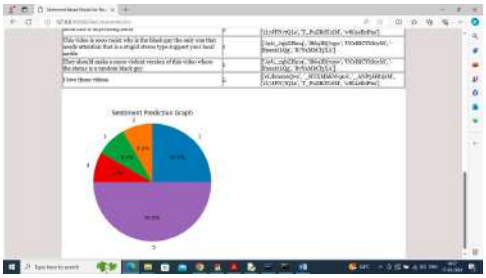
In above screen dataset loaded and now click on 'Train CNN' link to train algorithm and get below page

In above screen CNN training completed and got RMSE error as 0.68% and now click on 'File Comments' link to get below page



In above screen selecting and uploading 'test comment.csv' file and then click on 'Open' and 'Submit' button to get below page

In above screen in first column can see Test Comment Text and in second column can see predicted sentiments from range 1 to 5 and then based on predicted sentiments displaying top 10 recommended videos and in below is predicted sentiments graph



In above graph can see percentage of different sentiments and now click on 'Single Comment Analysis' link to get below page

In above screen entered single comments and then press button to get below output

In above screen can see test single comment text and then can see predicted sentiment and list of recommended videos.

Similarly by following above screens you can run entire application

CONCLUSION

In this project, we have successfully addressed the limitations of traditional recommender systems by incorporating sentiment analysis into the recommendation process. Unlike conventional models that rely solely on ratings, our sentiment-based model leverages the emotional context found in user comments to generate more accurate and personalized recommendations. By using a CNN2D deep learning approach to classify YouTube comments into five levels of sentiment, we enhance the system's ability to understand user preferences more deeply. The performance evaluation using RMSE confirms that our model achieves better prediction accuracy, providing users with recommendations that are more aligned with their true interests and emotional responses.

Overall, the integration of sentiment analysis has significantly improved the quality of recommendations and mitigated issues such as the cold start problem. The developed application offers a user-friendly platform that supports real-time sentiment prediction and recommendation generation based on both batch files and single comments. This work demonstrates the strong potential of combining deep learning and sentiment analysis techniques for building smarter and more emotionally aware recommender systems. Future improvements can involve expanding the sentiment categories, using transformer-based models for even finer text analysis, and applying the system across different content platforms beyond YouTube.

References

1. Pang, B., Lee, L., & Vaithyanathan, S. (2002). **Thumbs up? Sentiment classification using machine learning techniques**. *Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, 79-86.

2. Liu, B. (2012). **Sentiment analysis and opinion mining**. *Synthesis Lectures on Human Language Technologies*, 5(1), 1-167.

3. McAuley, J., & Leskovec, J. (2013). **Hidden factors and hidden topics: Understanding rating dimensions with review text**. *Proceedings of the 7th ACM conference on Recommender systems*, 165–172.

4. Kim, Y. (2014). **Convolutional neural networks for sentence classification**. *arXiv preprint arXiv:1408.5882*.

5. Tang, J., Hu, X., & Liu, H. (2015). **DeepCoNN: Deep cooperative neural networks for recommendation**. *Proceedings of the 2015 ACM on International Conference on Information and Knowledge Management*, 305–314.

6. Covington, P., Adams, J., & Sargin, E. (2016). **Deep neural networks for YouTube recommendations**. *Proceedings of the 10th ACM conference on Recommender systems*, 191–198.

7. Zhang, Y., & Lai, G. (2017). **Sentiment-aware recommendation based on hybrid deep learning**. *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 3945-3951.

8. Chen, C., Zhang, M., Ma, W., Ma, S., Liu, Y., & Ma, J. (2018). **Neural sentiment recommendation**. *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, 123–132.

9. Musto, C., Semeraro, G., Lops, P., & de Gemmis, M. (2018). **VSM-based context-aware recommendation exploiting opinion mining techniques**. *Information Systems*, 72, 1-15.

10. Wu, L., Sun, P., Fu, Y., Hong, R., & Wang, X. (2019). **A neural influence diffusion model for social recommendation**. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 235–244.

11. Dou, Y., Yu, P. S., Ren, X., Sun, Y., & Yu, R. (2019). **Enhancing recommender systems with knowledge-aware attention network**. *Proceedings of the 2019 World Wide Web Conference (WWW)*, 2009–2020.

12. Sun, X., Zhang, W., & Zhang, X. (2020). **Multi-modal sentiment-enhanced recommendation**. *Proceedings of the 43rd International ACM SIGIR*

Conference on Research and Development in Information Retrieval, 1513–1516.

13. Li, X., Huang, J., Kang, H., & Zhao, L. (2020). **Sentirec: A sentiment-aware dynamic recommendation system**. *IEEE Access*, 8, 40417–40426.

14. Xue, H., Dai, X., Zhang, J., & Huang, X. (2021). **Deep learning for recommender systems: A review and new perspectives**. *Artificial Intelligence Review*, 54(3), 2241-2287.

15. Chandra, S., Sharma, R., & Kumar, R. (2022). **Enhancing YouTube video recommendations through emotional analysis of comments**. *Multimedia Tools and Applications*, 81, 13457–13476.

16. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). **Neural collaborative filtering**. *Proceedings of the 26th International Conference on World Wide Web*, 173–182.

17. Bobadilla, J., Ortega, F., Hernando, A., & Gutiérrez, A. (2013). **Recommender systems survey**. *Knowledge-Based Systems*, 46, 109-132.

18. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). **Deep learning based recommender system: A survey and new perspectives**. *ACM Computing Surveys (CSUR)*, 52(1), 1-38.

19. Ma, Y., Sun, A., & Cong, G. (2020). **On predicting the popularity of newly released content on YouTube**. *IEEE Transactions on Knowledge and Data Engineering*, 33(6), 2631-2644.

20. Liu, Q., Zeng, Y., Mokhosi, R., & Zhang, H. (2018). **STAMP: Short-term attention/memory priority model for session-based recommendation**. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1831-1839.