iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** III **Month of publication:** March 2022

DOI: https://doi.org/10.22214/ijraset.2022.40753

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue III Mar 2022- Available at www.ijraset.com

Quantization of Product using Collaborative Filtering Based on Cluster

E. Sankar[1], L. Karthik[2], Kuppa Venkatasriram Sastry[3]
[1]*Assistant Professor, Dept. Of CSE, SCSVMV (Deemed to be University), Kanchipuram, TamilNadu, India*
[2, 3]*Student, Dept. Of CSE, SCSVMV (Deemed to be University), Kanchipuram, TamilNadu, India*

Abstract: ***Because of strict response-time constraints, efficiency of top-k recommendation is crucial for real-world recommender systems. Locality sensitive hashing and index-based methods usually store both index data and item feature vectors in main memory, so they handle a limited number of items. Hashing-based recommendation methods enjoy low memory cost and fast retrieval of items, but suffer from large accuracy degradation. In this paper, we propose product Quantized Collaborative Filtering (pQCF) for better trade-off between efficiency and accuracy. pQCF decomposes a joint latent space of users and items into a Cartesian product of low-dimensional subspaces, and learns clustered representation within each subspace. A latent factor is then represented by a short code, which is composed of subspace cluster indexes. A user's preference for an item can be efficiently calculated via table lookup. We then develop block coordinate descent for efficient optimization and reveal the learning of latent factors is seamlessly integrated with quantization. In this paper we also propose similarity method that has the ability to exploit multiple correlation structures between users who express their preferences for objects that are likely to have similar properties. For this we use a clustering method to find groups of similar objects.***
Index Terms: Product Quantization, Clustering, Product Search, Collaborative Filtering.

I. INTRODUCTION

APPROXIMATE nearest neighbour (ANN) search in a static database has achieved great success in supporting many tasks, such as information retrieval, classification and object detection. However, due to the massive amount of data generation at an unprecedented rate daily in the era of big data, databases are dynamically growing with data distribution evolving over time, and existing ANN search methods would achieve unsatisfactory performance without new data incorporated in their models. In addition, it is impractical for these methods to retrain the model from scratch for the continuously changing database due to the large scale computational time and memory. Therefore, it is increasingly important to handle ANN search in a dynamic database environment.

ANN search in a dynamic database has a widespread applications in the real world. For example, a large number of news articles are generated and updated on hourly/daily basis, so a news searching system [1] requires to support news topic tracking and retrieval in a frequently changing news database. For object detection in video surveillance [2],video data is continuously recorded, so that the distances between/among similar or dissimilar objects are continuously changing.

For image retrieval in dynamic databases [3], relevant images are retrieved from a constantly changing mage collection, and the retrieved images could therefore be different overtime given the same image query. In such an environment, real-time query needs to be answered based on all the data collected to the database so far.

In recent years, there has been an increasing concern over the computational cost and memory requirement dealing with continuously growing large scale databases, and therefore there are many online learning algorithm works [4], [5],[6] proposed to update the model each time streaming data coming in. Therefore, we consider the following problem. Given a dynamic data base environment, develop an online learning model accommodating the new streaming data with low computational cost for ANN search.

Recently, several studies on online hashing [7], [8], [9],[10], [11], [12], [13] show that hashing based ANN approaches can be adapted to the dynamic database environment by updating hash functions accommodating news treaming data and then updating the hash codes of the exiting stored data via the new hash functions.

Searching is performed in the Hamming space which is efficient and has low computational cost. However, an important problem that these works have not addressed is the computation of the hash code maintenance. To handle the streaming fashion of the data, the hash functions are required to be frequently updated, which will result in constant hash code re computation of all the existing data in the reference database.

©IJRASET: All Rights are Reserved

This will inevitably incur an increasing amount of update time as the data volume increases. In addition, these online hashing approaches require

The system to keep the old data so that the new hash code of the old data can be updated each time, leading to inefficiency in memory and computational load. Therefore, computational complexity and storage cost are still our major concerns in developing an online indexing model.

Product quantization (PQ) [14] is an effective and successful alternative solution for ANN search. PQ partitions the original space into a Cartesian product of low dimensional subspaces and quantizes each subspace into a number of sub-code words. In this way, PQ is able to produce a large number of code words with low storage cost and perform ANN search with inexpensive computation. Moreover, It preserves the quantization error and can achieve satisfactory recall performance.

Most importantly, unlike hashing-based methods representing each data instance by a hash code, which depends on a set of hash functions, quantization based methods represent each data instance by an index, which associates with a codeword that is in the same vector space with the data instance. However, PQ is a batch mode method which is not designed for the problem of accommodating streaming data in the model. Therefore, to address the problem of handling streaming data for ANN search and tackle the challenge of hash code re computation, we develop an online PQ approach, which updates the code words by streaming data without the need to update the indices of the existing data in the reference database, to further all eviate the issue of large scale update computational cost.

II. MODULE

A. *Admin Module*

In this module, the Admin has to login by using valid user name and password. After login successful he can perform some operations such as List all users and authorize, Register with News channel name and login, Add News Categories, Set news quantization date, Select category and add news, all news post and give option to update and delete, List all news post by quantization, List All News Posts by clusters

based on news cat, List All Users News transactions by keyword, View online product quantization by chart, View all news post rank in chart.

B. *User Module*

In this module, there are n numbers of users are present. User should register before performing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user can perform some operations like View your profile, Search news by content keyword, select hash code to show all news titles, Show all your search transactions based on keyword and hash code.

III. SYSTEM ANALYSIS

A. *Existing System*

1) Existing hashing methods are grouped in data-independent hashing and data-dependent hashing. One of the most representative work for data-independent hashing is Locality Sensitive Hashing (LSH) [20], where its hashing functions are randomly generated. LSH has the theoretical performance guarantee that similar data instances will be mapped to similar hash codes with a certain probability. Since data-independent hashing methods are independent from the input data, they can be easily adopted in an online fashion.

2) Data-dependent hashing, on the other hand, learns the hash functions from the given data, which can achieve better performance than data independent hashing methods. Its representative works are Spectral Hashing (SH) [19], which uses spectral method to encode similarity graph of the input into hash functions, IsoH [18] which finds a rotation matrix for equal variance in the projected dimensions and ITQ [17] which learns an orthogonal rotation matrix for minimizing the quantization error of data items to their hash codes.

3) To handle nearest neighbour search in a dynamic database, online hashing methods have attracted a great attention in recent years. They allow their models to accommodate to the new data coming sequentially, without retraining all stored data points. Specifically, Online Hashing, Adapt Hash and Online Supervised Hashing are online supervised hashing methods, requiring label information, which might not be commonly available in many real-world applications. Stream Spectral Binary Coding (SSBC) [9] and Online Sketching Hashing (OSH) are the only two existing online unsupervised hashing methods which do not require labels, where both of them are matrix sketch-based methods to learn to represent the data seen so far by a small sketch.

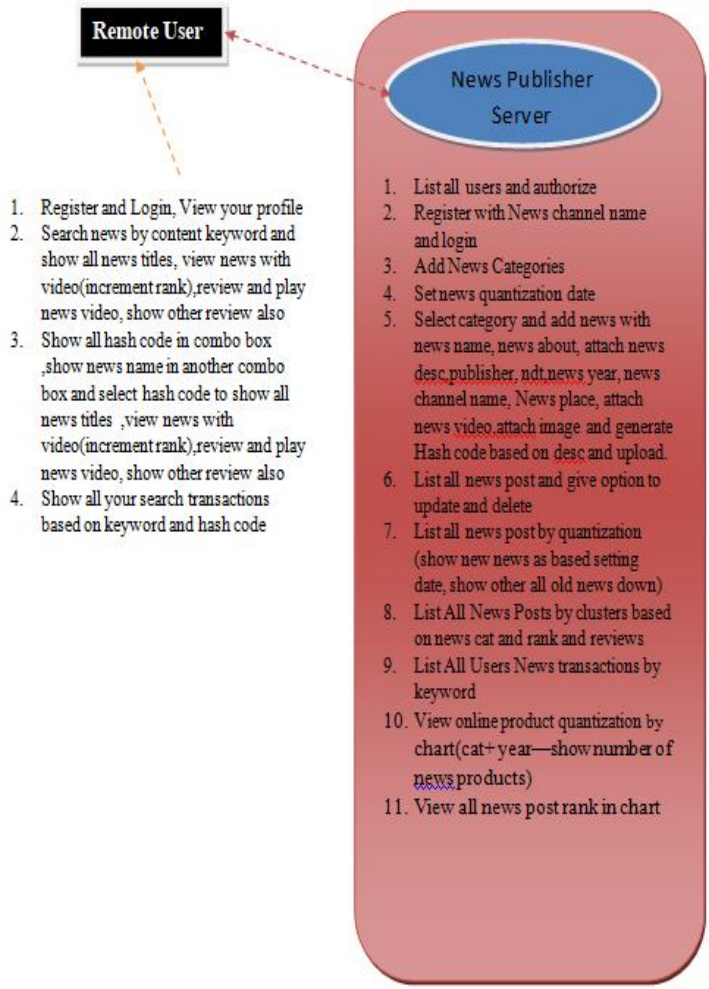
4) However, all the online hashing methods suffer from the existing data storage and the high computational cost of hash code maintenance on the existing data. Each time new data comes, they update their hash functions accommodating to the new data and then update the hash codes of all stored data according to the new hash functions, which could be very time-consuming for a large scale database.

B. Proposed System

1) The quantization error and can achieve satisfactory recall performance. Most importantly, unlike hashing-based methods representing each data instance by a hash code, which depends on a set of hash functions, quantization based methods represent each data instance by an index, which associates with a codeword that is in the same vector space with the data instance.

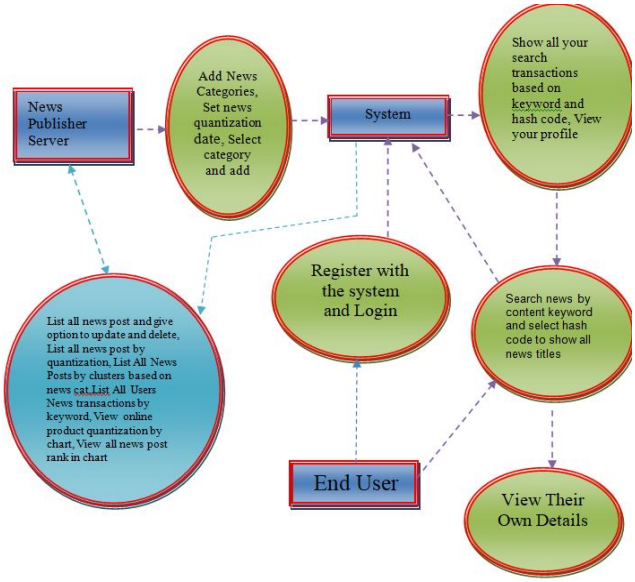
2) However, PQ is a batch mode method which is not designed for the problem of accommodating streaming data in the model. Therefore, to address the problem of handling streaming data for ANN search and tackle the challenge of hash code re computation, the system develops an online PQ approach, which updates the code words by streaming data without the need to update the indices of the existing data in the reference database, to further alleviate the issue of large scale update computational cost.

IV. ARCHITECTURE

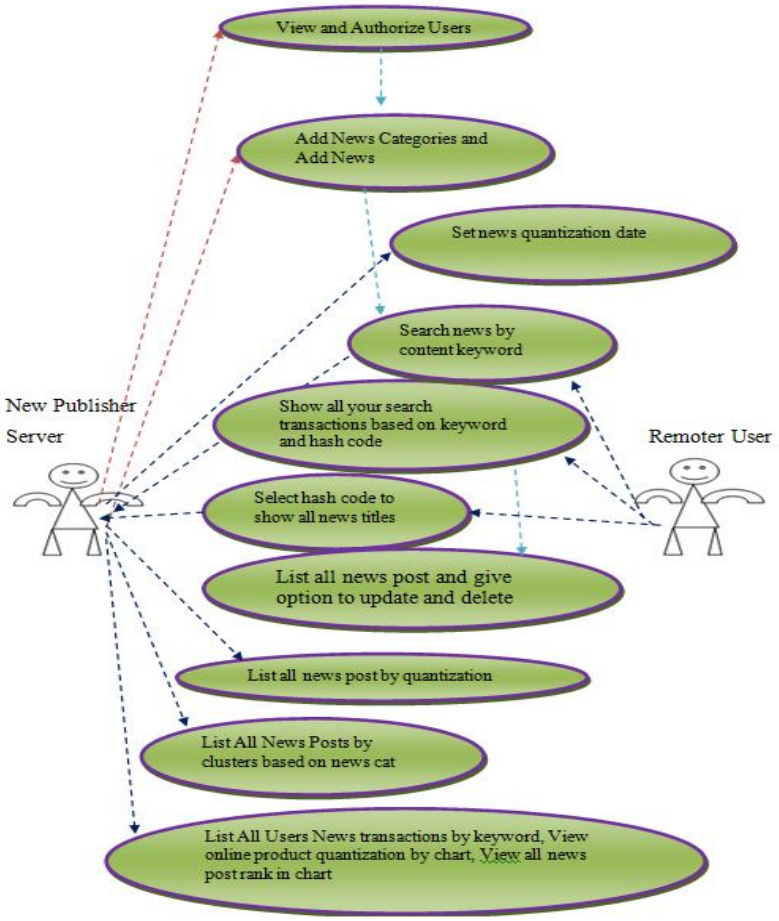


The architecture shows how the admin and user Login into the system in a easy way. And also it Show the news which is outdated and which is current trending in a easy way. The admin canadd the news in which category he want. And also the user search the news in which he want in a easy way.

V. DATA FLOW



VI. USE CASE DIAGRAM

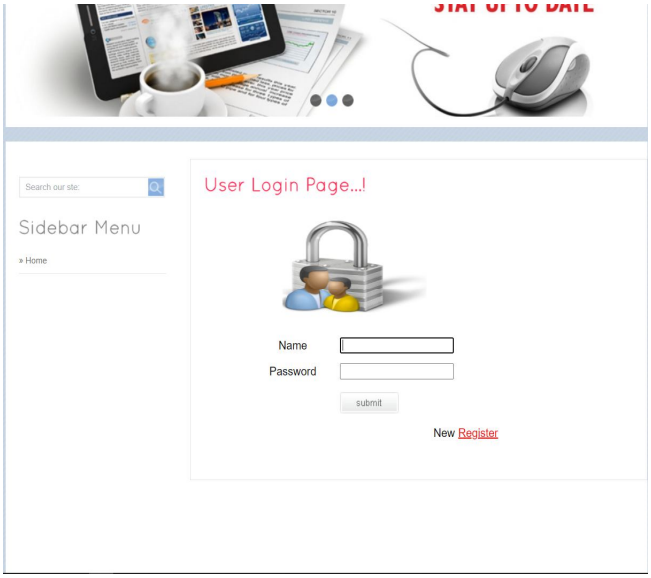


VII. OUTPUT SCREENSHOTS

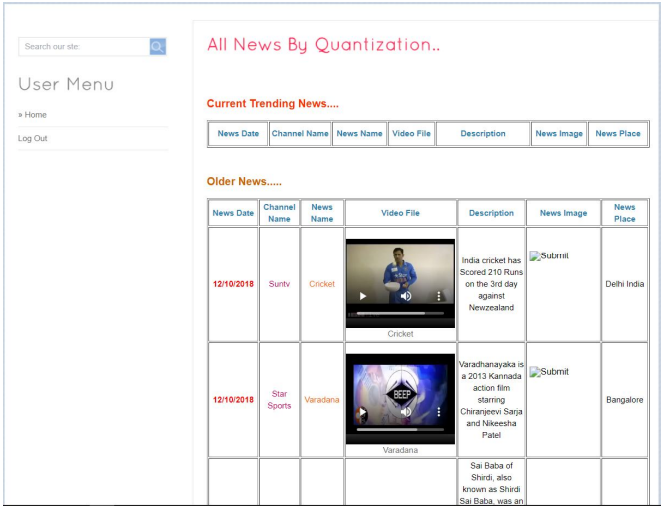
A. *Home Page*



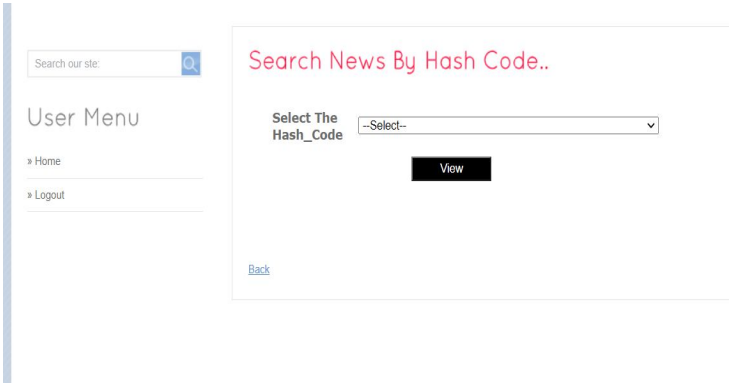
B. *Login Page*



C. *Current News Page*



D. Search News by Hash Code

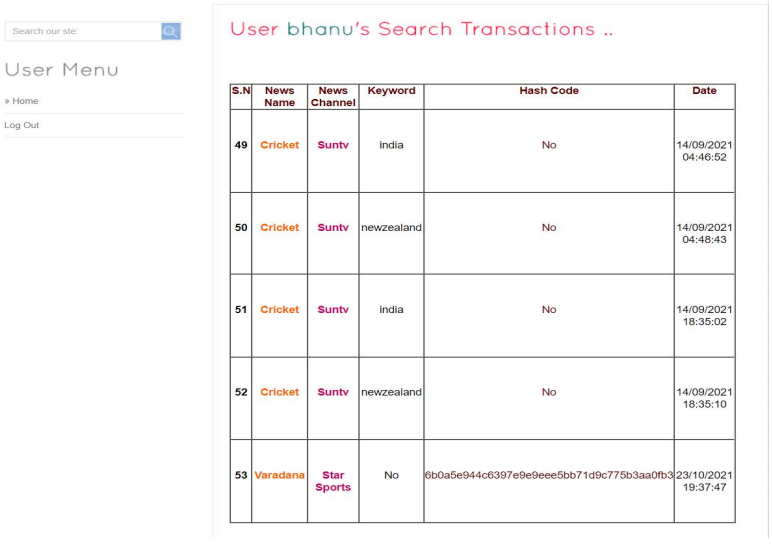


E. Videos Separated by Categories





F. View All Search Transaction Page



VIII. CONCLUSION

In this paper, we have presented our online PQ method to accommodate streaming data. In addition, we employ two budget constraints to facilitate partial codebook update to further alleviate the update time cost. A relative loss bound has been derived to guarantee the performance of our model. In addition, we propose an online PQ over sliding window approach, to emphasize on the real-time data. Experimental results show that our method is significantly faster in accommodating the streaming data, out performs the competing online hashing methods and unsupervised batch mode hashing method in terms of search accuracy and update time cost, and attains comparable search quality with batch mode PQ. The user can search the news in which category he/she want in easy without any difficult way. The videos are divided into categories by using clustering technique.

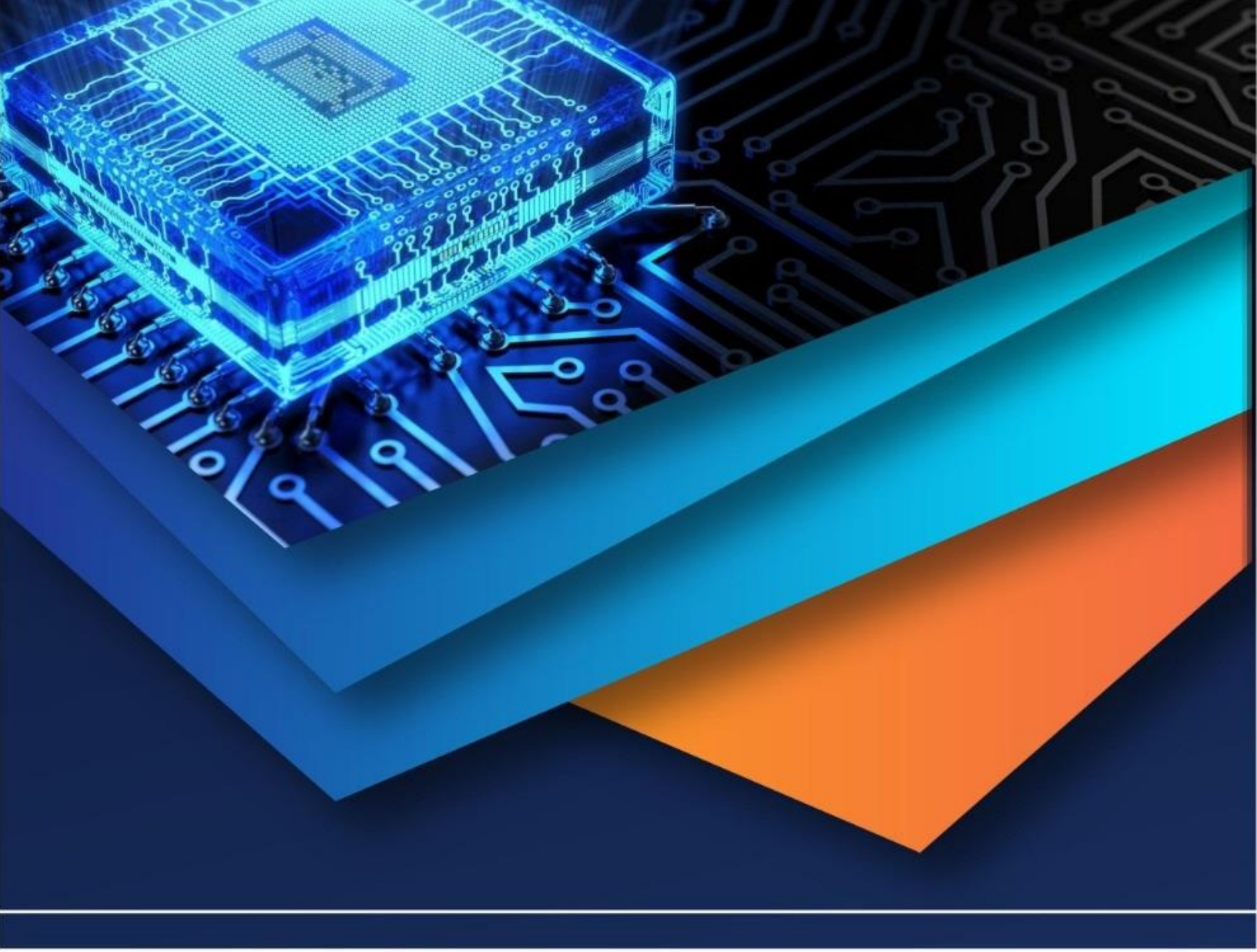
IX. FUTURE ENHANCEMENTS

This application avoids difficulty concern with it.

It is an easy obtain to search the videos in which category user like. Well I and my team member have worked hard in order to present an improved website better than the existing one's regarding the information about various activities. Still we found that the project can be done in a better way. Primarily, when we search the videos we will all outdated and current trending videos we will see in all are combined together while using this website we can see which video is currently trending and which video is outdated. And also we can see the videos are arranged according to the video concept. These are two enhancements that we could think at present.

REFERENCES

[1] Liu, M., Dai, Y., Bai, Y., & Duan, L.-Y. (2020). Deep Product Quantization Module for Efficient Image Retrieval. ICASSP 2020 – 2020 IEEE International Conference on \ Acoustics, Speech and Signal Processing (ICASSP).
[2] Lian, D., Xie, X., Chen, E., & Xiong, H. (2020). Product Quantized Collaborative Filtering. IEEE Transactions on Knowledge and Data Engineering, 1–1.
[3] Gupta, M., Thakkar, A., Aashish, Gupta, V., & Rathore, D. P. S. (2020). Movie Recommender System Using Collaborative Filtering. 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC).
[4] Zarzour, H., Jararweh, Y., & Al-Sharif, Z. A. (2020). An Effective Model-Based Trust Collaborative Filtering for Explainable Recommendations. 2020 11th International Conference on Information and Communicati-on Systems(ICICS).
[5] Pu, X., & Zhang, B. (2020). Clustering collaborative filtering recommendation algorithm of users based on time factor. 2020 Chinese Control And Decision Conference (CCDC).
[6] Veras De Sena Rosa, Ricardo Erikson; Guimaraes, Felipe Augusto Souza; Mendonca, Rafael da Silva; Lucena, Vicente Ferreira de (2020). Improving Prediction Accuracy in Neighborhood-Based Collaborative Filtering by Using Local Similarity. IEEE Access, 8(), 142795–142809.
[7] Bindra, K., & Mishra, A. (2017). A detailed study of clustering algorithms. 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO). doi:10.1109/icrito.2017.8342454
[8] M. Ghashami and A. Abdullah, "Binary coding in stream," CoRR,vol.abs/1503.06271, 2015.
[9] C. Li, W. Dong, Q. Liu, and X. Zhang, "Mores: online incrementalmultiple-output regression for data streams," CoRR, vol.abs/1412.5732, 2014.
[10] C. Leng, J. Wu, J. Cheng, X. Bai, and H. Lu,"Online sketchinghashing," in CVPR, 2015, pp. 2503–2511.
[11] F. Cakir and S. Sclaroff, "Adaptive hashing for fast similaritysearch," in ICCV, 2015, pp. 1044–1052.
[12] Q. Yang, L. Huang,W. Zheng, and Y. Ling, "Smart hashing updatefor fast response," in IJCAI, 2013, pp. 1855–1861.
[13] F. Cakir, S. A. Bargal, and S. Sclaroff, "Online supervised hashing,"CVIU, 2016.
[14] H. J´egou, M. Douze, and C. Schmid, "Product quantization fornearestneighbor search," PAMI, vol. 33, no. 1, pp. 117–128, 2011.
[15] M. Norouzi and D. J. Fleet, "Minimal loss hashing for compactbinary codes," in ICML, 2011, pp. 353–360.
[16] W. Liu, J. Wang, R. Ji, Y. Jiang, and S. Chang, "Supervised hashingwith kernels," in CVPR, 2012, pp. 2074–2081.
[17] Y. Gong and S. Lazebnik, "Iterative quantization: A procruste an approach to learning binary codes," in CVPR, 2011, pp. 817–824.
[18] W. Kong and W. Li, "Isotropic hashing," in NIPS, 2012, pp. 1655–1663.
[19] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in NIPS,2008, pp. 1753–1760.
[20] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in highdimensions via hashing," in VLDB, 1999, pp. 518–529.
[21] A. Babenko and V. S. Lempitsky, "Additive quantization for extreme vector compression," in CVPR, 2014, pp. 931–938.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)