Mobile App Usage Pattern prediction using Hierarchical Flexi-Ensemble Clustering (HFEC) for Mobile Service Rating

P. Priyanga (priyangaphd26@gmail.com)
Alagappa University https://orcid.org/0000-0001-7179-0466
A. R. Nadira Banu Kamal
Mohamed Sathak Hamid College of Arts and Science for Women

Research Article

Keywords: Data mining, pattern prediction, app behavior analysis, mobile service rating, ensemble clustering, genetic algorithm, customer rating

Posted Date: February 26th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-272906/v1

Version of Record: A version of this preprint was published at Wireless Personal Communications on October 8th, 2021. See the published version at https://doi.org/10.1007/s11277-021-09048-0.

Mobile App Usage Pattern prediction using Hierarchical Flexi-Ensemble Clustering (HFEC) for Mobile Service Rating

*[1]P. Priyanga, [2]A. R Nadira Banu Kamal

[1]Research Scholar, Alagappa University, Karaikudi, Tamilnadu, India

[2]Principal, Mohamed Sathak Hamid College of Arts and Science for Women, Ramanathapuram, Tamilnadu, India.

*Corresponding Mail ID: priyangaphd26@gmail.com.

***Abstract*—**

Background: Nowadays, the mobile app market becomes rapidly increased in world wide. The mobile app marketers have smart enough to understand the requirements and demands of customers and perform their aspirations. They delight them. It provides growth, profitability, and creativity with lot of inventions. The main aim of this research is to analyze the customer interest and preferences of mobile service providers.

Methodology: This paper proposed the clustering model named as Hierarchical Flexi-Ensemble Clustering (HFEC). It provides the final result with robustness and improved quality. Before clustering, the unwanted features are removed by using the Genetic Algorithm based on the Collective Materials (GACM) technique. The customer preferences are analyzes with the clustering of mobile usage patterns.

Results: The analysis determined that the app usage pattern based on the most frequent word, rating category, rating character count, rating word count and content-based rating in the google play store app dataset. Finally, the results are compared with the existing methods to analyze the superior performance of proposed method. The comparison analysis is estimated based on the based on the average hit rate at different cache sizes.

Conclusion: The work is concluded with the app pattern prediction in the form of clustering for app marketing service. From the marketing side, they can analyze the customer preferences and satisfaction.

***Index Terms*— Data mining, pattern prediction, app behavior analysis, mobile service rating, ensemble clustering, genetic algorithm, and customer rating.**

I. INTRODUCTION

The smartphone becomes human comrades due to its flexible services and the number of increased mobile applications accelerates the rate of smartphone adoption. Whenever the user searches apps in the google play store, the list of available apps shown to the user along with its app name and user rating. At the end of 2018, there have been more than 5.6million applications present in the Google play store and Apple play store. Furthermore, the increased number of mobile apps will enhance installs or downloads. Generally, the users like to download the apps which are the highly-rated apps because it rectifies the other user's satisfaction than the rest of the applications. The number of popular apps had billions of active users and downloads. With these million number of popular apps, one of the important challenges is to catch the focus of users. To increase the quality of the app and users' high rating, the companies and developers utilized some strategies and techniques such as (i) directly allowing users to provide the five-star rating, (ii) describing free text formation, (iii) request new features, etc…The app store market contains some characteristics such as books, music, entertainment, games, and movies, and so on. Also, they had the opportunity to make the changes and add the new features in the new release app based on the user's reviews and feedback ratings. In the mobile app market, the developers involved accessing the crowd sourced information about their developed apps such as app reviews, app usage, ratings, and app relevant posts. To increase the partialities of mobile users, apps quality, and user experiences, the developers should understand how people manage mobile applications in day by day. Various researchers analyzed the app usage pattern which is based on the data traffic, number of unique users, N umber of downloads, and length of network access. [1] suggested the correlation analysis to derive the app usage pattern. The correlation between app usage and device model was derived and it's important to determine the various users' behavior. [2] analyzed the temporal behavior of users with mental scale analysis. The clustering method was used for characterizing the archetypical user engagement. [3] introduced the Spatio-temporal event detection approach and then the clustering is performed based on contextual data. It enhances the retrieval information by mitigating searching space and searching of related clusters. [4] demonstrated the matrix factorization method to detect the app usage pattern and user intents.

Data mining involves the mining of app usage information in two pleats. Firstly, it learns the app usage pattern of users than can improve the quality of the app. Secondly, it extracts the user profiles and suggests high relevant apps to users. The widely used data mining techniques are sequential pattern discovery,

association discovery, clustering, classification, and forecasting. The app marketing can be enhanced with the mining of user feedback, this analysis helped to learn how the users think about the app and their recommendations to develop the app. With the usage pattern, the app store market recommends apps based on their preferences to users. [5] [6] focused to analyze the mobile user's preferences of over mining app management activities. Then they depicted the behavioral pattern from those activities for detecting more accurate user preferences.

The existing methodologies contain the following limitations:

- Inefficient of prediction results
- Most of the prediction system only consider the location
- More time consumption

The aim of this research is to identify customer preferences and satisfaction in the form of mobile app usage patterns. To achieve this the proposed system utilized the clustering strategy in a highly efficient way. This system is used to understand the development and customer preferences of various mobile service providers. Additionally, we have analyzed the app usage pattern analysis in terms of most frequent words, rating category, rating character count, rating word count, and content-based rating. We developed the implementation to cluster customer satisfaction based on their preferences. Here, the clustering of the accuracy level is enhanced by using the similarity index and merge score. The time taken for determining the cluster position is very less. The expected outcomes of this research will be the patterns to get the satisfaction of customers for any mobile services. The motivation of this research is defined as below,

- To achieve efficient feature selection by a Genetic Algorithm based on the Collective Materials (GACM) approach.
- To evaluate the app usage pattern by Hierarchical Flexi-Ensemble Clustering (HFEC) technique.
- To increase the efficiency of the clustering method by novel similarity indexing formula.
- To effectively cluster the customer satisfaction based on analysis of mobile app usage pattern.

Organization of the paper:
The remaining portion of this paper is scheduled as follows: Section II demonstrates the various clustering methods to determine the mobile application pattern. Section III illuminates about the proposed Hierarchical Flexi-Ensemble Clustering (HFEC) technique for determining the mobile application usage pattern in google play store. Section IV exhibits the performance measures of the proposed system and Section V concludes the proposed work.

II. REVIEW OF LITERATURE WORK

This section described the various existing works of clustering methods to determine the mobile application pattern. Also, we discussed the popularity prediction of mobile apps with their behavior.

[7] prepared the crowd listening method for the release planning of mobile apps. The mobile apps marketing becomes bigger and it is anticipated to grow over 100 billion dollars in 2020. To achieve this successful development, the developers and competitive environment require to create and manage the high-quality apps with its new features. The application marketplaces allow users to provide reviews and comments. These reviews aim to achieve the recommending apps between enormous users and also provide useful information for developers and it acts the precious information for reporting failures and proposing new features. The developers can able to access the users to review manually to analyses the source of information. This research proposed the CLAP (Crowd Listener for release Planning) technique to assist the developers to achieve this task based on the web application with three categories such as,

1. Characterizes the user reviews based on their comment information
2. Clustering based on their related reviews.
3. Prioritizing of cluster reviews

They evaluated the various steps in the CLAP process and it shows the highest accuracy in characterizing and clustering reviews. This working tool applied to industrial environments applications.

[8] suggested the popularity prediction model for understanding mobile application usage. The highest growth of network services and smart devices drive mobile application development. Some of the popular applications enhance the network capacity and user experiences such as BSs, Apps, etc… However, that is critical to analyses the app usage pattern and traffic consumption through BSs in the metropolitan area. Through the network interfaces, the mobile big data was collected and make it easy for the data-driven approach in features characterizing. This research proposed the edge catching strategy based on the app's characteristics, POIs - points of interest, traffic, logs that are generated by various apps categories. They examined the various apps' temporal characteristics and then analyzed the logs and then traffic generated by various BSs clusters. It predicted the top N popular apps in a certain period through different clusters. It is more beneficial to network operators to analyses the different apps traffic distribution.

[9] utilized the feature extraction process for analyzing important features. Google play store platform allows giving their reviews about the app. The user reviews and comments help to retrieve the requirements of potential apps. There were various researchers mentioned the mobile app feature extraction in user reviews. The important feature extraction is the most important challenge and to recover this issue, this research proposed finding collocation for extracting the mobile app features in reviews and extracting the infrequent features from reviews based on the rules of extraction. Then, it compared the similarity of overall retrieved features from reviews with app descriptions. The similarity measurement technique contains the similarity of 1. Single term by the corresponding feature of each term, 2. Synonym refers to WordNet synsets, 3. Sentence based on an estimation of cosine similarity and lexical-semantic

vector. The results were proved with the performance metrics of precision and recall.

[10] proposed the fine-grained mobile app clustering model with Retrofitted Document Embedding. To analyses the clusters automatically with no predefined categories this research initializes the clusters in terms of tittle keywords and then combines the similar clusters. The proposed method differentiates the accurate clustering step with tittles to improve the clustering model performance. In the result section, the tagged set was evaluated when the processing of the accurate clustering step. This tagged set was further used for learning a high-performance document vector. The evaluation results that the proposed method of performances which was decreased 1.18 entropy value and increased 0.19 accuracy value compared to k-means clustering and SVM algorithm.

[11] proposed the user behavior clustering algorithm for mobile applications. The analysis of user behavior clustering divides users into multiple groups. This research proposed the two collections of algorithms such as the fuzzy clustering algorithm and the two-layer clustering algorithm with user behavior data. The traditional first-layer clustering method develops the DBSCAN algorithm that exchanges the neighborhood radius with the designed similarity of user session and it has optimized clusters merge condition. Then retrieve the user sessions feature vectors and utilized the enhanced FCM algorithm. This method initializes the membership matrix to speed up the weighted value and convergence speed to solve the local optimum issues. The results of both algorithms provide effective performances and these clustering methods applied to various analysis scenarios.

[12] explicated the two-layer clustering method for mobile application analysis. This new framework provides a macro perspective on mobile CRM applications. The mobile application marketers used data analysis to achieve their company products and services to sell to targeted customers in terms of accurate marketing. They track the group structure changes periodically by using the established clustering model. This research only investigated the clustering with data usage behavior, mobile voice, customer base data, and customer contributions. It does not design to increase the grouping of customer variable selection function.

[13] predicted the app's popularity with retention rates and trend filters. Commonly, the popularity of the mobile app measured by installations, downloads and user ratings. The challenge of these measures that they provide indirect usage. The retention rates define several users to continue with app installation that have been recommended to determine the app life-cycles. They conducted a large scale of usage trends and retention rates on app usage dataset from more than 213, 667 apps and 339,842 users. The analysis showed 65% of application loss of their users in the first week and 35% of application loss for the top 100 applications.

[14] Suggested the clustering-based mining textual features to provide the benefits of developers and users. In-app store, for the mobile apps of latent clustering they proposed the novel method for similarity calculation based on claimed behavior. In the proposed system, the features are extracted by using ontological analysis. The retrieved attributes were used to clustering the app with agglomerative hierarchical clustering. They evaluated the proposed method of 17,877 apps from Google app stores and BlackBerry in 2014. The proposed system results improved the quality of existing categorization from 0.02 to 0.41 for blackberry and from 0.03 to 0.21 for google stores. They also determined the strong Spearman rank correlation as ρ= 0:99 for blackberry and ρ= 0:96 for google among a various number of apps.

[15] proposed the community-based diffusion method with spectral clustering and Markov chain for mobile social networks (MSN). The information exchange becomes an important challenge in emerging MSN. This research addressed the issue of determining the top-k influential users which means the users spread the information effectively in the network. To reduce the problem of the spreading period, this paper used the k-center problem that has NP-hard time complexity. In the end, they selected top-k influential patterns in each community. The NS-2 simulation performed the proposed method of results in MSNs.

[16] Described a clickstream tool for modeling online user behavior. Online services are utterly dependent on user involvement. Either online social networks or crowd sharing sites, it is important but difficult to understand user behavior. The partitioning technique influences iterative tuning to capture hierarchy within user clusters and to produce intuitive functionality to visualize and captured user behavior. Service providers can examine dominant user habits and classifications as a summary while visualizing fine-grained behavior patterns across each category with the aid of the visualization tool. The tool just not require prior knowledge or consideration about user groups, so it can easily capture unexpected or previously unknown behaviors. The efficacy was shown through the case studies on two large-scale online networks. This tool reliably detected suspicious behaviors and even predicted potential user activities. Finally, resources and identification are shared with the whisper data science team while waiting for more detailed documents, the basic comments were extremely positive.

[17] introduced a CHABADA approach to effectively predict the applications whose behavior was unexpectedly provided their description. Several instances of false misleading ads have been found, a new effective detector for existing unknown malware had been acquired as a side effect. Just as mining software archives had been opened up new chances for empirical software engineering, the mining application and their specifications open up many new chances for automated natural-language research. As a result, the proposed gained a range of perceptions into the Android app environment that calls for an action. The main application vendors require to be much clearer about what their applications do to earn their money. App store vendors such as Google should adopt better standards to rescue deceptive or inaccurate ads. Then the way that Android asks its users for permission is disabled. Finally, CHABADA is designed to point out and illustrated discrepancies that should be easier to observe.

[18] [14]Proposed a novel methodology that evaluates app similarity which is based on claimed behavior. While categorizing software systems concerning their functionality leads to many advantages for both users as well as developers Characteristics are extracted using information retrieval supplemented by ontological analysis and used as attributes to describe apps. Those attributes are then used to cluster applications using agglomerative hierarchical clustering. This

methodology tested 17,877 applications mined from blackberry and google app store. As a result, the approach has slightly enhanced the existing categorization quality for both blackberry and google stores. Here it is also found that a good spearman ranks the correlation for google and blackberry between the number of apps and the proper granularity with increasing size. Eventually, there is a positive correlation between the mean score given to the raters and the finest granularity. Also, a feature extraction and clustering approach will be explored to allow reverse engineering tasks in domain analysis and also to compare different clustering of app stores with different similarity steps.

[19] Proposed a met heuristics-based clustering ensemble method. For clustering ensembles, this analysis performed an improved generation process and co-association matrix in the co-occurrence method. The key component analysis is used to enhance efficiency. The main issue is the mobile application. The marketing strategy for the actual application is therefore based on the better outcome.

[20] analyzed the customer satisfaction of mobile app by using data mining methods. Currently, various data mining techniques had been used to investigate customer data. This research showed that the number of data mining techniques to be applied in the analysis of customer satisfaction. The machine learning methods were applied with CRISP-DM methodology on the dataset. Some of the modeling techniques are Naïve Bayes, logistic regression and decision tree. The predictive performance was the most essential to analyses the customer like about the app. The results are achieved 90% accuracy and the feature selection methods improve the overall accuracy as well as negative class precision.

[21] provided data mining applications for pattern recognition. In telecommunication, mobile subscriber's effects from the data traffic every day. In network, the data traffic provides certain characteristics of behavior. The data mining applications help to analyses the data traffic features. This paper proposed the new technology in the form of exponential binning of data preprocessing to minimize the noise and smoothening of data. Then used the k-means algorithm for clustering the data traffic stream and mining the behavior characteristics of subscribers from clusters.

[22] analyzed the mobile application usage and predict location based on cluster. App usage prediction and smartphone user's location are important problems in the current researches. Some of the smartphone sensors are defined as GPS, accelerometer, gyroscope, microphone, camera, and bluetooth, which makes it easy to analyses the user behavior data for specific analysis. But the number of apps increases and user behavior differences predicted a challenging task. The proposed work conducted the dataset with 30000 users from a leading IT company in China that converts the data into frequency, recency and monetary variables and finally performed clustering analysis to analyses the user behavior. For every cluster, the predicted models are developed by using the training dataset and testing dataset.

[23] This research investigates the young children's real varied app on large aggregated Australian datasets in primary schools. The dataset contains 15,000 Android devices over three years. The association analysis and clustering analysis have been employed to analyses the usage app patterns in data mining methods. The evaluation results showed the five distinct app use of patterns. The various use patterns of implications were discussed about teaching and learning.

[24] utilized the data clustering methods for predicting the temporal data characteristics in various mobile applications through wireless communications. The existing researches focused only on the analysis of mobile traces, and call records. This research concentrates on the usage of mobile applications to characterize and detect their behavior. They utilized the mobile application usage logs to characterize the mobile applications of temporal behavior in-network service provider. It showed that the utilization of classes to analyses the future usage of specific mobile applications via distance calculation and similarity comparison techniques.

[25] provided the two popular clustering algorithms such as fuzzy c-means and k-means. Clustering is the most suitable method to identify the hidden groups in large datasets. The research conducted on a mobile app dataset with 7196 applications that were clustered by using proposed clustering methods. After the various techniques of pre-processing such as standardization and outlier removal, the clustering algorithms are run with various parameters to reach the highest performances and optimal values. The results proved that the fuzzy c-means algorithm provides the highest quality compared to the k-means algorithm. [26] determined the various applications of usage behavior through different kinds of smartphones. The traditional researches only analyzed the data from smartphone-based user reports, this research involves the analysis of elementary characteristics of smartphone users. The research conducted on the dataset with 106,762 Android users and determined 382 distinct types of users based on their usage behavior by using the feature ranking selection technique and two-step clustering technique.

III. PROPOSED WORK

A novel method named Hierarchical Flexi-Ensemble Clustering (HFEC) for determining the mobile application usage pattern in google play store environment. Additionally, the novel feature selections strategy was utilized to select the more relevant features from a pre-processed dataset. There are various types of information present in app stores from the developer's side such as app description, app features, downloads, ratings, comments, etc…The prediction of mobile app usage pattern is the most important for mobile service rating. To achieve this the proposed novel methods used in a highly efficient way with three types of a process named as

- Pre-processing
- Feature selection by using GACM
- Pattern prediction by using HFEC

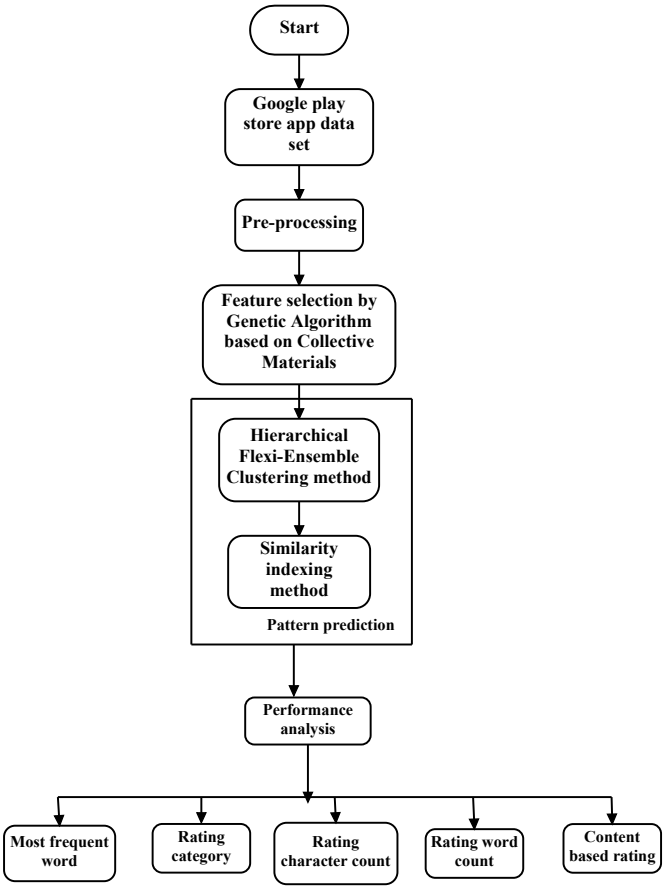

Figure 1: Overall Flow of the proposed system

3.1. Pre-processing:

The initial dataset contains missing values, failure to load, corrupt data, incomplete extraction, and uninformative parts and lots of noises. The pre-processing process defined as cleaning of text and data for further processing and it improves the quality of data. To improve the proposed method of performance, the noises should be reduced from the dataset. Initially, determine the missing values, null values, and noises from the dataset. Then handling the null values by dropping the rows or columns. To handle the missing values, we achieved from the calculation of mean, median of the feature and replace it with the missing values. After pre-processing, the developed dataset contains app id, app name, category, app review, app rating, type, price, genres, last updated version, current version, and android version. Once the dataset pre-processed completely, then it can be used for further processing.

3.2. Feature selection: Genetic Algorithm based on Collective Materials (GACM)

After pre-processing the dataset, the feature selection process will be done by using a genetic algorithm that is based on collective materials. The dataset contains a large number of features and has to optimize the most important features. The independent dataset does not utilize to create the predictive model, so we required to minimizing the error of the model. Only the relevant features provide important information of output, as well as irrelevant features, consist of a minimum amount of information regards output. To achieve this more efficiently, the Genetic algorithm is used based on collective materials. The main advantage of this approach is, it can handle the large dataset with many features. The optimization problem solving is to determine those input features that consist of an enormous amount of information about output. To measure the important information of random variables, the entropy and collective materials are introduced in this research. Here, we utilized a new formula for computing the conditional collective materials between the candidate feature and given a subset of features in the local search process. Generally, entropy measured the uncleared random variables. The joint entropy of two discrete random variables are defined as,

$G(Y,X) = -\sum_{y\epsilon\Im}\sum_{x\epsilon\wp} Q(y,x)\log_q(y,x)$

The conditional entropy defined as

$G(Y|X) = -\sum_{y\epsilon x}\sum_{x\epsilon y} Q(y,x)\log_q(y|x)$

The collective material defined as the common information of two random variables x and y. a large number of collective materials between two random variables are closely related. If the value of collective material is zero, then the two variables are unrelated. For discrete random variables, the entropy and collective material can be estimated as,

$G(Y) = -\int Q(y) log Q(y) dy$

$J(Y;X) = \iint Q(y,x) log \frac{q(y.x)}{q(y).q(x)}$ dydx

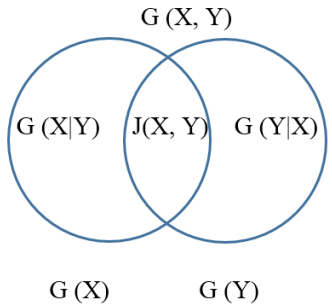

Figure 2: Relation between collective material and entropy

The evaluation of collective material between the discrete variables defined as,

$J(Y;X) = \hat{Q}(X) - \hat{Q}(Y|X)$

Then the conditional collective materials are represented as,

$J(C; f_i | f_s) = J(C; f_i) - \{I(f_s; f_i)\} - I(\{(f_s; f_i)\} | C)$

The local and global search of feature selection are done by maximizing collective material.

Calculation of cluster position by using GACM:

Discrete random variable x has y alphabets with its probability density function represented as q(y) = Qs{Y=y}, y ϵ $\Im$ then the entropy of Y can be defined as

$G(Y) = -\sum_{y\epsilon\Im} q(y)\log q(y)$

For the case of two discrete random variables, i.e., Y and X, the joint entropy of Y and X is defined as follows:

$G(Y,X) = -\sum_{y\epsilon x}\sum_{x\epsilon y} Q(y,x)\log_q(y,x)$

Where q(y, x) denotes the joint probability density function of Y and X. the remaining uncertainty can be described by the conditional entropy, which is defined as

$G(Y|X) = -\sum_{y\epsilon\Im}\sum_{x\epsilon\wp} Q(y,x)\log_q(y|x)$

The common information of two random variables Y and X is defined as the collective material between them,

$$J(Y;X) = \sum_{y\epsilon\Im}\sum_{x\epsilon\wp} Q(y,x)\log\frac{q(y.x)}{q(y).q(x)}$$

The relation between the collective material and the entropy can be described in Eqs. (5)–(7),

$J(Y;X) = G(Y) - G(Y|X) = G(X) - G(X|Y)$

$= G(Y)+G(X)\vdash G(Y,X)$

$J(Y;X) = J(X;Y)$

$J(Y;Y) = G(Y)$

For the case of continuous random variables, the differential entropy and collective material can be defined as

$G(Y) = -\int Q(y)logQ(y)dy$

$J(Y;X) = \iint Q(y,x)log\frac{q(y.x)}{q(y).q(x)}\,dydx$

$\hat{Q}(Y= j) = \frac{\sum_{i=1}^{m} h_{ji}}{M}$

$\hat{Q}(X= i) = \frac{\sum_{j=1}^{n} h_{ji}}{M}$

$\hat{Q}(Y= j|X= i) = \frac{h_{ji}}{\sum_{j=1}^{n} h_{ji}}$

$\hat{Q}(Y) = -\sum_{j=1}^{n}\hat{Q}(Y=j)\log(\hat{Q}(Y=j))$

$\hat{Q}(Y|X=i) = -\sum_{j=1}^{n}\hat{Q}_{ji}\log\hat{Q}_{ji}$

$\hat{Q}(Y|X) = \sum_{i=1}^{m}\hat{Q}(X = i)\, G(Y|X=i)$

$J(Y;X) = \hat{Q}(X) - \hat{Q}(Y|X)$

$I = \max\max J(X;X_e)$ eq33

$Q_f(e) <= \frac{1}{2}(G(X)-1(X:X_e))$ eq34

where Qf(e) is the misclassification probability of a classifier.

$Q_f(e) >= \frac{G(X)-J(X;X_e)-g(Q_f(e))}{\log(|\varsigma|-1)}$

$Q_f(e) >= \frac{G(X)-J(X;X_e)-1}{\log(|\varsigma|}$

$Q(B) = \frac{\sum_{j=1}^{r} M_{jj}}{M}$ and $Q(F) = \sum_{j=1}^{r}\frac{M_{j*}}{M}\cdot\frac{M_{*j}}{M}$

$T = \frac{Q(B)-Q(F)}{1-Q(F)}$

3.3. Pattern prediction: Hierarchical Flexi-Ensemble Clustering (HFEC)

The dataset contains the various characteristics of app usage patterns and the proposed clustering approach effectively supports the search of relevant patterns. Previously, the app usage pattern was analyzed based on the data traffic, number of unique users, number of downloads, and length of network access. In this research, the proposed clustering model analyzed the important variables or pattern prediction that influences the overall rating of the mobile application.

After the feature selection, the clustering is done by using the proposed Hierarchical Flexi-Ensemble Clustering (HFEC) method. Ensemble clustering defines the situation that the number of different (input) clustering has obtained for a particular dataset and it is requested to determine the single (consensus) clustering which is a better fit in some sense than the existing clustering. It provides the final result with robustness and improved quality. In this proposed method, there are multiple options present to structure the clustering process. Initially, we calculate the distance between two clusters and the cluster center is computed. Finally, the similarity between those clusters is calculated with the novel framework. We handle the clusters and ensemble members differently. Each ensemble contains a single instance and it is a part of one cluster at all times. The simple layout algorithm is employed to determine the cluster position or cluster center based on the upper limit and lower limit computation. The upper limit xhi and lower limit xlo are computed for each cluster. The layout is parameterized with vitals such as horizontal spacing, line thickness of ensemble member, minimum and maximum vertical spacing between clusters. Then the similarity matrix θ defined as the similarity between two data pints u and v. If the similarity between two data points is 1, that is assigned to the same cluster or else 0. The base

clustering (A) for all data matrices defined as Y' and the similarity is calculated by,

$$\theta_{vu} = \frac{\sum_{q=1\ldots B} \sum_{p=1\ldots A} \theta_{vu}(\pi_p(Y'q))}{B \times A}$$

The proposed flexible ensemble clustering method solves the clustering problem and dendrogram selection problems.

1. *B* Base of exponential decrease of argument ∇
2. *B* to meet the global parameters ∇min and ∇max
3. a pow ((∇max/∇min), (1/*m*levels 1))
4. function COMPUTE X_HI(*u*, x_{lo}, ∇)
5. *Qu*.get Points From Current TimeStep()
6. if *Q* = 0 then
7. *B* Cluster does not contain any point at this time step
8. return x_{lo}∇*a*
9. end if ∇
10. *u*.$x_{lo}$$x_{lo}$
11. if *u* is a leaf then
12. *B* End recursion: consider the members in this cluster
13. $x_{hi}$$x_{lo}$ + ∇+ line Width *Q* +∇
14. *B* Vertical center of the point with rank 0
15. off x_{lo} +∇+ lineWidth/2
16. for each *q* in *Q* do
17. *q*. *x* off + line Width *q*. rank
18. end for
19. return x_{hi}
20. end if
21. x_{hi} of the rightmost descendant of *v*.left
22. tmp ← compute X_hi (*u*.left, x_{lo} +∇, ∇/*a*)
23. *v*.x_{hi} ← compute X_hi (*u*.right, tmp +Δ, ∇/*a*) + ∇
24. return *u*.x_{hi}
25. end function
26. compute X_hi (root, 0, ∇max)

The proposed HEFC method allows interacting with an ensemble on the complete scale from the overall mean-field to individual members.

4. RESULTS AND DISCUSSION

After estimating all the proposed models and methods, the results are estimated. This unit depicts the various performance measures of the proposed clustering model with the most frequent word, rating category, rating character count, rating word count and content-based rating. The analysis of pattern prediction was estimated with these measures.

4.1. Dataset description:

The proposed method is evaluated by using google play store apps dataset from Kaggle [27]. This dataset consists of 10,840 applications of ranking and reviews information. It contains multiple variables such as app name, app id, category, reviews, rating, No of installs, app size, app type, content rating, price, genres, last updated version, current version, and android version. The dataset variables are detailed as below table 1.

Table 1: Representation of dataset variables

1	Category_name	Each uploaded app associated with its name in google play store
2	No_of_reviews	Each app had the number of reviews
3	App_size	App size is mentioned in the app description and it is associated with its name
4	No_of_installs	Each app mentioned with the number of installs eg, 1k+, 2.5m, etc…
5	Typ_of_app	The app type contains free and paid

6	Price_of_app	The paid app associated with its pricing
7	Content_rating	Each app had the content rating eg, 1+, 13+, etc…
8	Genres	Genres associated with its app and it is different from the category
9	Andoid_version	To install the app, the minimum android version is required
10	Word_count_in_name	Overall words presents an in-app title that is computed for each app
11	Symbol_count_in_name	Total number of the symbols presents an in-app title that is computed for each app
12	Character_count_in_name	Total number of characters presents an in-app title that is computed for each app
13	Category_related	For each app, the Boolean variable is calculated and it matches the number of words used in-app with the number of words used in category
14	Free_in_title	For each app, the boolean variable is calculated and it determines whether the word used the in-app title or not.
15	Digits_in_tittle	Boolean variable which determines if any numeric value presents in-app title or not
16	Year_in_title	A Boolean variable that determines the year used in-app title.

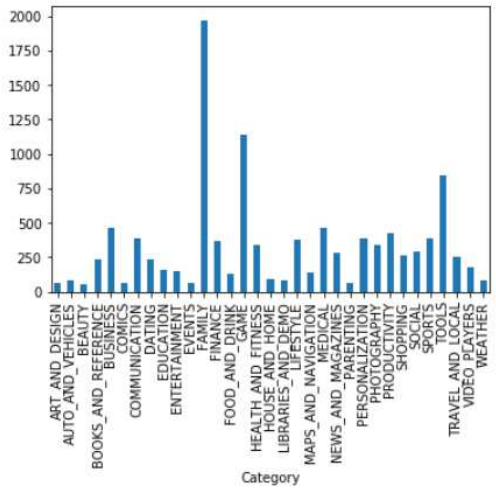

Figure 3: various category ratings

The mobile application that is used nowadays can be divided into many categories such as art and design, family, comics, communication, game, maps, social, sports, shopping and so on. The figure 3 shows the rating given based on the different categories of application available to the users. Application are also installed by several number of users every day. The app with better review will be installed by many users. The figure 4 explains the installation count of application by the number of users.

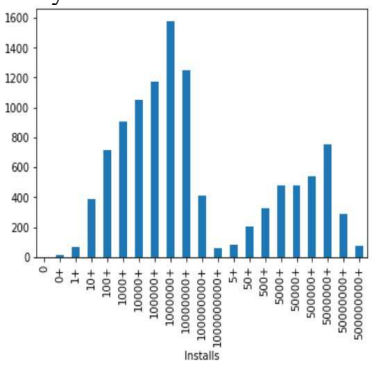

Figure 4: Count based on app's installed

4.2. Pattern prediction analysis:

In pattern prediction analysis, the most frequent word, rating category, rating character count, rating word count and content-based rating were analyzed. The proposed cluster model clustered the mobile application pattern with important variables. The developers of various application are introduced to users by their names. Some of the commonly used word that appear in many application include app, mobile, live, pro, chat and many more. In figure 5, the count of most frequently used word in the application name among the various applications available in internet. From the figure, word app is the most frequently used by the application developers.

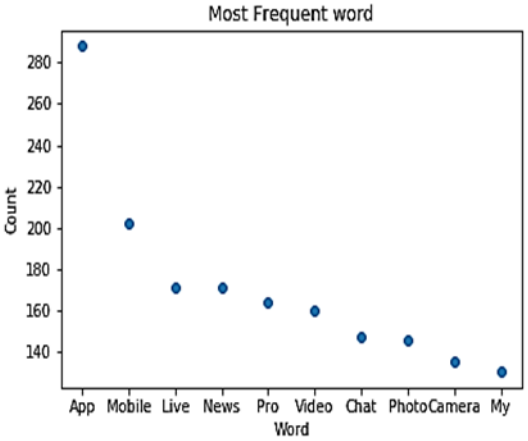

Figure 5: Most frequent word used in app name and its count based on usage

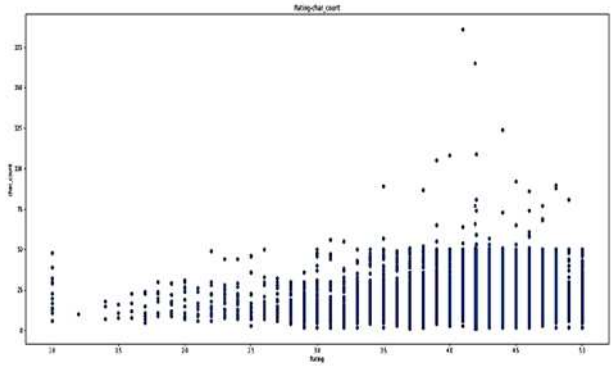

Figure 6: Ratings based on character count

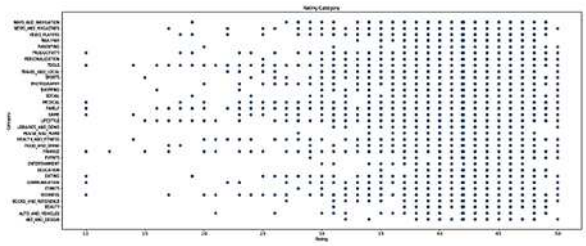

Figure 7: Ratings to different categories of app

The figure 6 shows the rating of the application based on their character count. Character count obtained rating in the range between 3.5 and 4.5. Rating of 1 was given at rare times. In figure 7, the rating obtained based on the different category of application is shown. It explained the rating given by app users to the various category. Some of the category of apps like beauty, sports, education, shopping, maps and many more never got lowest rating. At the same time, some app categories like communication, medical, family, tools and few other obtained least rating.

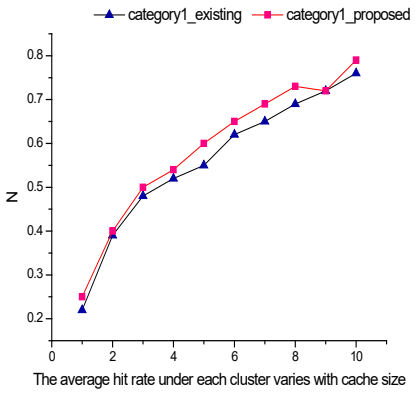

Figure 8: Comparison of 1st category of proposed and existing

Figure 8 represents the comparison between the existing and proposed of first category with respect to the average rate of hit with respect to the separate sizes of cache.

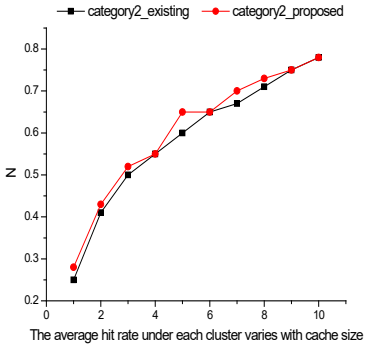

Figure 9: Comparison of 2nd category of proposed and existing

The average rate of hit changes with respect to the size of cache. The existing and the work proposed were compared and it is represented in the figure 9.

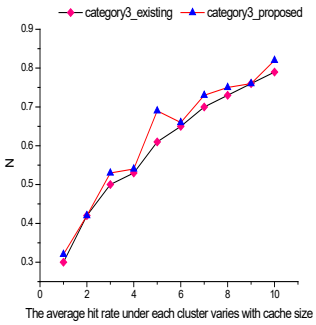

Figure 10: Comparison of 3rd category of proposed and existing

In figure 10, the third category of the proposed and existing are compared based on the average hit rate obtained according to various sizes of cache.

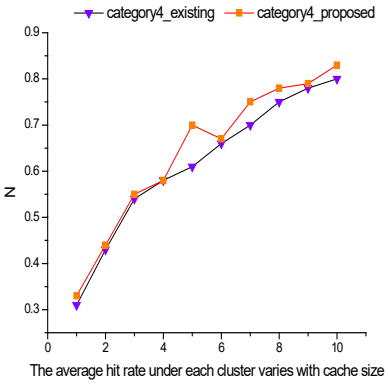

Figure 11: Comparison of 4th category of proposed and existing

The average rate of hit based on the different sizes of cache for the fourth category are compared and it is shown in figure 11. Also, for fifth and sixth categories the graph are represented in the figure 12 and figure 13 respectively.

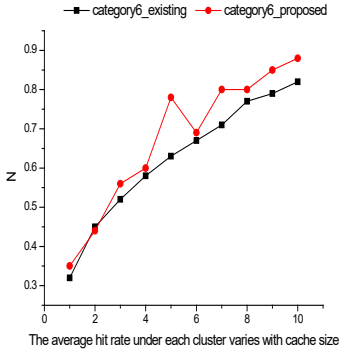

Figure 12: Fifth category comparison

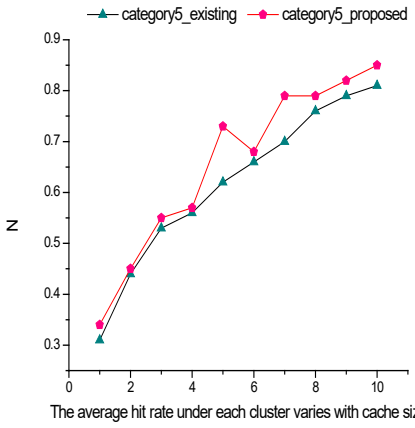

Figure 13: sixth category comparison

Figure 12 & 13: Comparison of 5th and 6th category of proposed and existing respectively

The models and method proposed were analyzed based on the most frequent word, character count ratings, ratings obtained for various categories are compared with graphical representations. From the results compared, the proposed models and the methods proved to be better than the existing approaches [8].

5. CONCLUSION

In this research, the detailed analysis of mobile app using patterns is carried out in google play store apps. The new framework of Hierarchical Flexi-Ensemble Clustering i.e., HFEC is proposed to influence app ratings. The general algorithm of GA based on the collective materials were used to remove the unnecessary features before clustering process. A novel formula based on similarity indexing were used to enhance the efficiency of clustering. The performance of the model proposed were evaluated by the considering the dataset taken from google play store. By comparison of the proposed model with existing based on the frequent word used in app, character count ratings, ratings given to various categories of app were illustrated with graph.

Later, considered about six categories for the proposed and the existing methods based on the average hit rate with respect to the different cache sizes are represented graphically. The main advantage of this approach is, it can handle the large dataset with many features. As a future work, sentiment analysis and image prediction can also be performed using highly efficient algorithms than the one used in this work.

CONFLICT OF INTEREST

- I confirm that this work is original and has either not been published elsewhere, or is currently under consideration for publication elsewhere.

Acknowledgements

- None.

Funding

- This research work was not funded by any organization/institute/agency.

Competing Interests

- None of the authors have any competing interests in the manuscript.

REFERENCES

[1] J. Lin and C. Dou, "A novel method for condition monitoring of rotating machinery based on statistical linguistic analysis and weighted similarity measures," *Journal of Sound and Vibration,* vol. 390, pp. 272-288, 2017.

[2] R. Bond, A. Moorhead, M. Mulvenna, S. O'Neill, C. Potts, and N. Murphy, "Exploring temporal behaviour of app users completing ecological momentary assessments using mental health scales and mood logs," *Behaviour & Information Technology,* vol. 38, pp. 1016-1027, 2019.

[3] R. Rawassizadeh, C. Dobbins, M. Akbari, and M. Pazzani, "Indexing multivariate mobile data through spatio-temporal event detection and clustering," *Sensors,* vol. 19, p. 448, 2019.

[4] Y.-N. Chen, M. Sun, A. I. Rudnicky, and A. Gershman, "Leveraging behavioral patterns of mobile applications for personalized spoken language understanding," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, 2015, pp. 83-86.

[5] X. Liu, W. Ai, H. Li, J. Tang, G. Huang, F. Feng*, et al.*, "Deriving user preferences of mobile apps from their management activities," *ACM Transactions on Information Systems (TOIS),* vol. 35, pp. 1-32, 2017.

[6] H. Li, W. Ai, X. Liu, J. Tang, G. Huang, F. Feng*, et al.*, "Voting with their feet: Inferring user preferences from app management activities," in *Proceedings of the 25th International Conference on World Wide Web*, 2016, pp. 1351-1362.

[7] S. Scalabrino, G. Bavota, B. Russo, M. Di Penta, and R. Oliveto, "Listening to the crowd for the release planning of mobile apps," *IEEE Transactions on Software Engineering,* vol. 45, pp. 68-86, 2017.

[8] M. Zeng, T.-H. Lin, M. Chen, H. Yan, J. Huang, J. Wu*, et al.*, "Temporal-spatial mobile application usage understanding and popularity prediction for edge caching," *IEEE Wireless Communications,* vol. 25, pp. 36-42, 2018.

[9] Q. Sutino and D. Siahaan, "Feature extraction from app reviews in google play store by considering infrequent feature and app description," in *Journal of Physics: Conference Series*, 2019, p. 012007.

[10] Y. C. Yoon, J. Lee, S. Y. Park, and C. Lee, "Fine-grained mobile application clustering model using retrofitted document embedding," *ETRI Journal,* vol. 39, pp. 443-454, 2017.

[11] Q. Su, Z. Jia, and L. Lu, "Research on user behavior clustering algorithm based on mobile application," *Journal of Intelligent & Fuzzy Systems,* vol. 35, pp. 1291-1300, 2018.

[12] P. AMARNATH and M. CHANDINI, "A Two-Layer Clustering Model for Mobile Customer Analysis," 2018.

[13] S. Sigg, E. Lagerspetz, E. Peltonen, P. Nurmi, and S. Tarkoma, "Exploiting usage to predict instantaneous app popularity: Trend filters and retention rates," *ACM Transactions on the Web (TWEB),* vol. 13, pp. 1-25, 2019.

[14] A. A. Al-Subaihin, F. Sarro, S. Black, L. Capra, M. Harman, Y. Jia*, et al.*, "Clustering mobile apps based on mined textual features," in *Proceedings of the 10th ACM/IEEE international symposium on empirical software engineering and measurement*, 2016, pp. 1-10.

[15] J. Ryu, J. Park, J. Lee, and S.-B. Yang, "Community-based diffusion scheme using Markov chain and spectral clustering for mobile social networks," *Wireless Networks,* vol. 25, pp. 875-887, 2019.

[16] G. Wang, X. Zhang, S. Tang, H. Zheng, and B. Y. Zhao, "Unsupervised clickstream clustering for user behavior analysis," in *Proceedings of the 2016 CHI*

Conference on Human Factors in Computing Systems, 2016, pp. 225-236.

[17] A. Gorla, I. Tavecchia, F. Gross, and A. Zeller, "Checking app behavior against app descriptions," in *Proceedings of the 36th International Conference on Software Engineering*, 2014, pp. 1025-1035.

[18] C. Schweitzer, "Mobile Phone Analysis through Clustering of Users based on Behavioral Features," Tilburg University, 2019.

[19] R.-J. Kuo, C. Mei, F. E. Zulvia, and C. Tsai, "An application of a metaheuristic algorithm-based clustering ensemble method to APP customer segmentation," *Neurocomputing,* vol. 205, pp. 116-129, 2016.

[20] J. Sunkpho and M. Hofmann, "Analyzing Customer Satisfaction of a Mobile Application using Data Mining Techniques," *Thammasat Review,* vol. 22, pp. 50-64, 2019.

[21] X. Wu, Y. Zhao, Q. Gu, and L. Gao, "Application of Data Mining for Behavior Pattern Recognition in Telecommunication," in *International Conference on Data Mining and Big Data*, 2018, pp. 426-433.

[22] X. Lu, B. Rai, Y. Zhong, and Y. Li, "Cluster-Based Smartphone Predictive Analytics for Application Usage and Next Location Prediction," *International Journal of Business Intelligence Research (IJBIR),* vol. 9, pp. 64-80, 2018.

[23] S. K. Howard, J. Yang, J. Ma, K. Maton, and E. Rennie, "App clusters: Exploring patterns of multiple app use in primary learning contexts," *Computers & Education,* vol. 127, pp. 154-164, 2018.

[24] K.-W. Lim, S. Secci, L. Tabourier, and B. Tebbani, "Characterizing and predicting mobile application usage," *Computer Communications,* vol. 95, pp. 82-94, 2016.

[25] T. r. C. Hakyemez, A. Bozanta, and M. Coşkun, "K-Means vs. Fuzzy C-Means: A Comparative Analysis of Two Popular Clustering Techniques on the Featured Mobile Applications Benchmark," 2019.

[26] S. Zhao, J. Ramos, J. Tao, Z. Jiang, S. Li, Z. Wu*, et al.*, "Discovering different kinds of smartphone users through their application usage behaviors," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 498-509.

[27] Dataset, "Google playstore apps dataset," 2019.

Figures

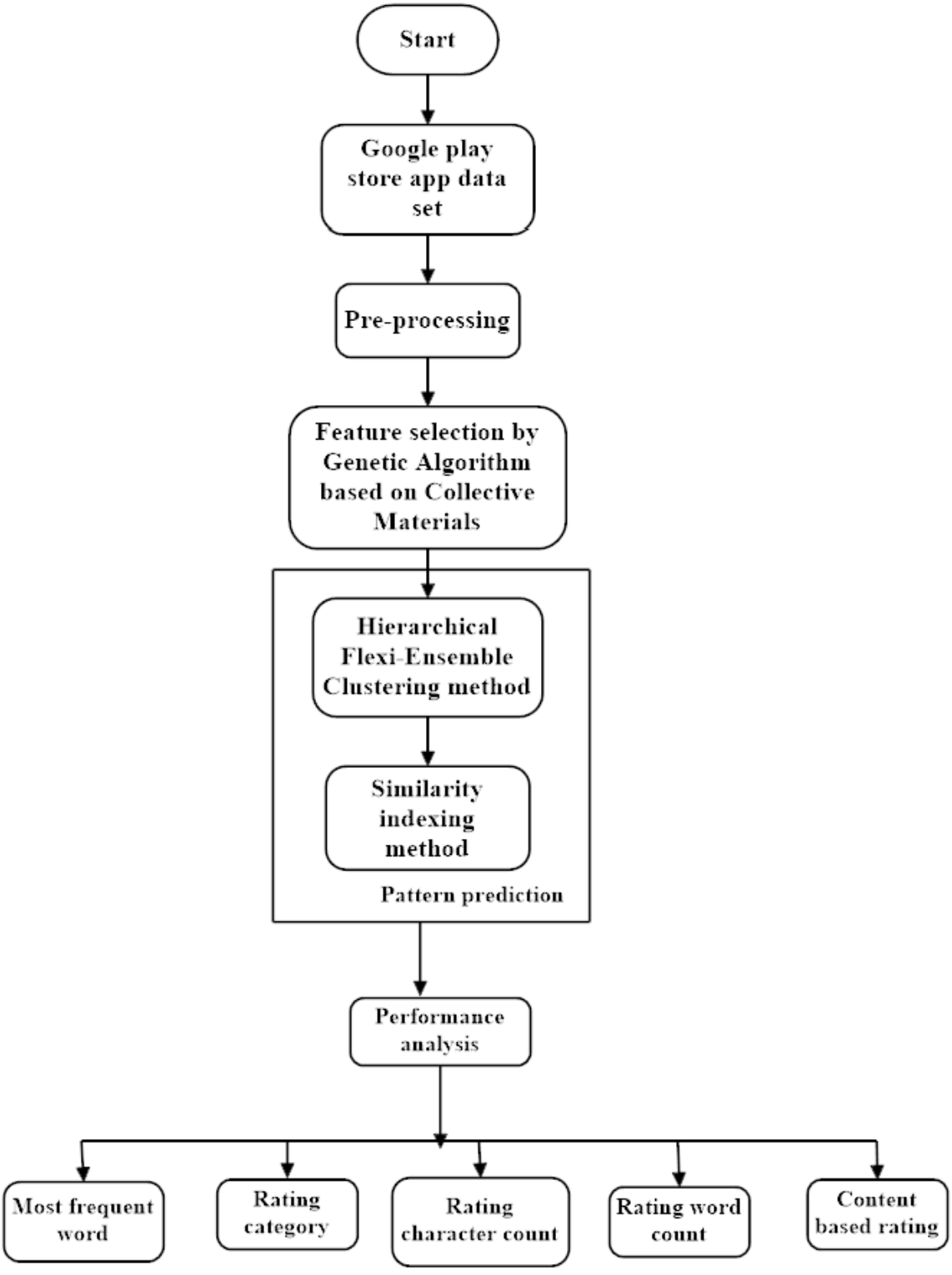

Figure 1

Overall Flow of the proposed system

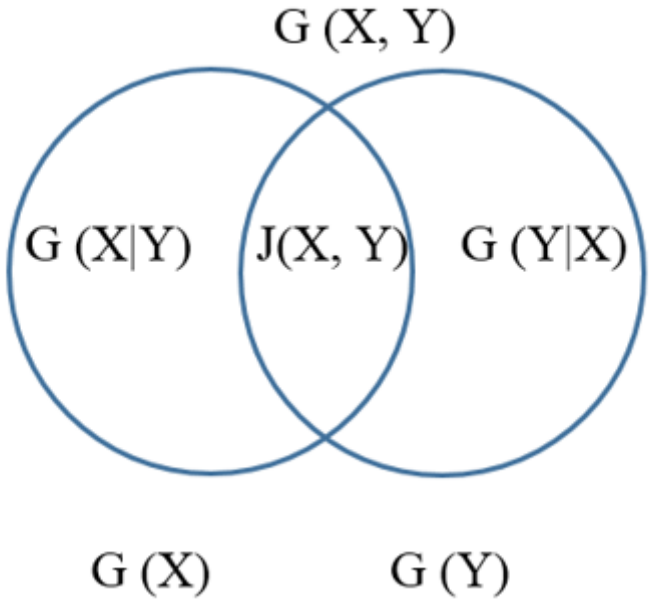

Figure 2

Relation between collective material and entropy

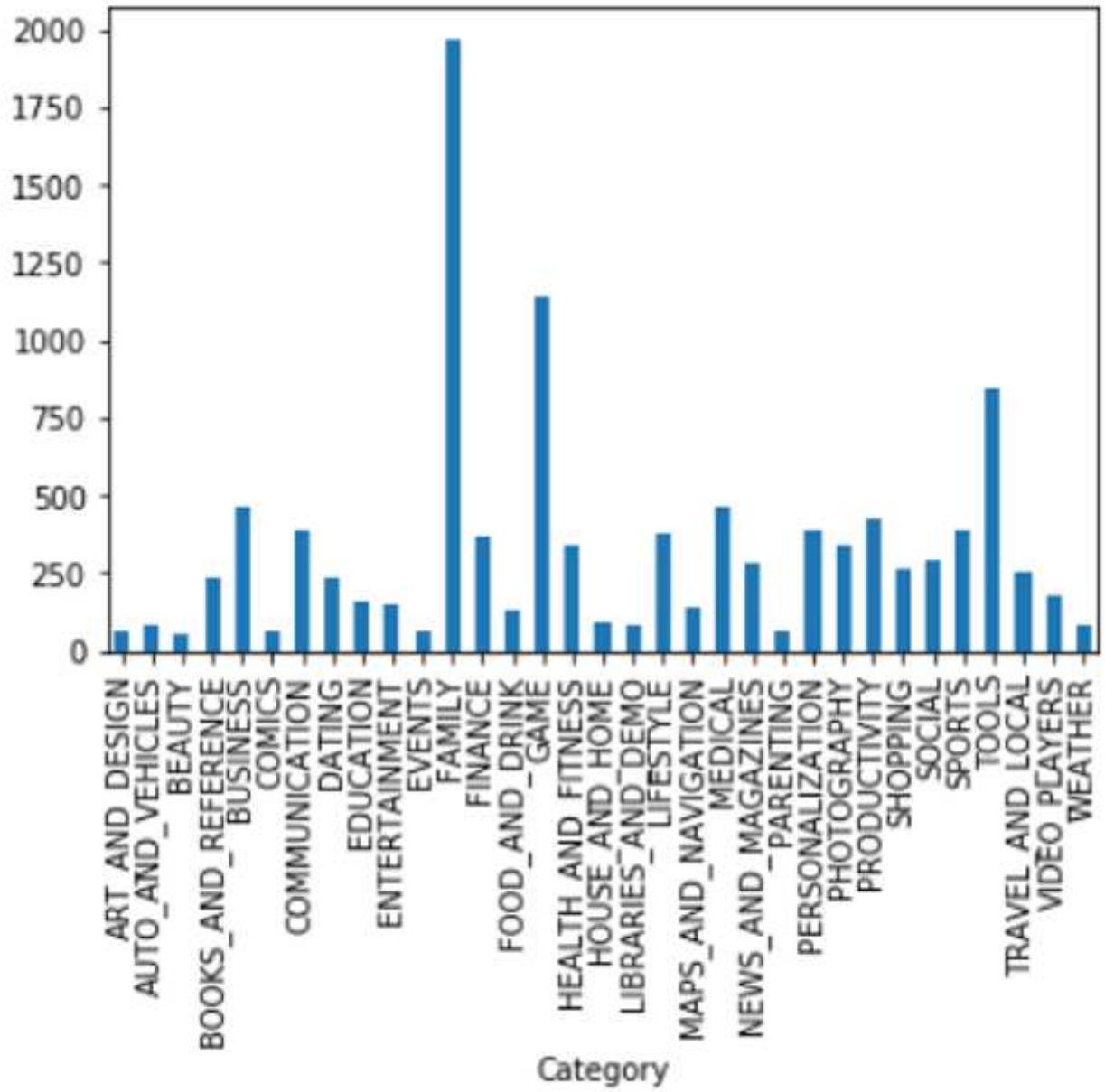

Figure 3

various category ratings

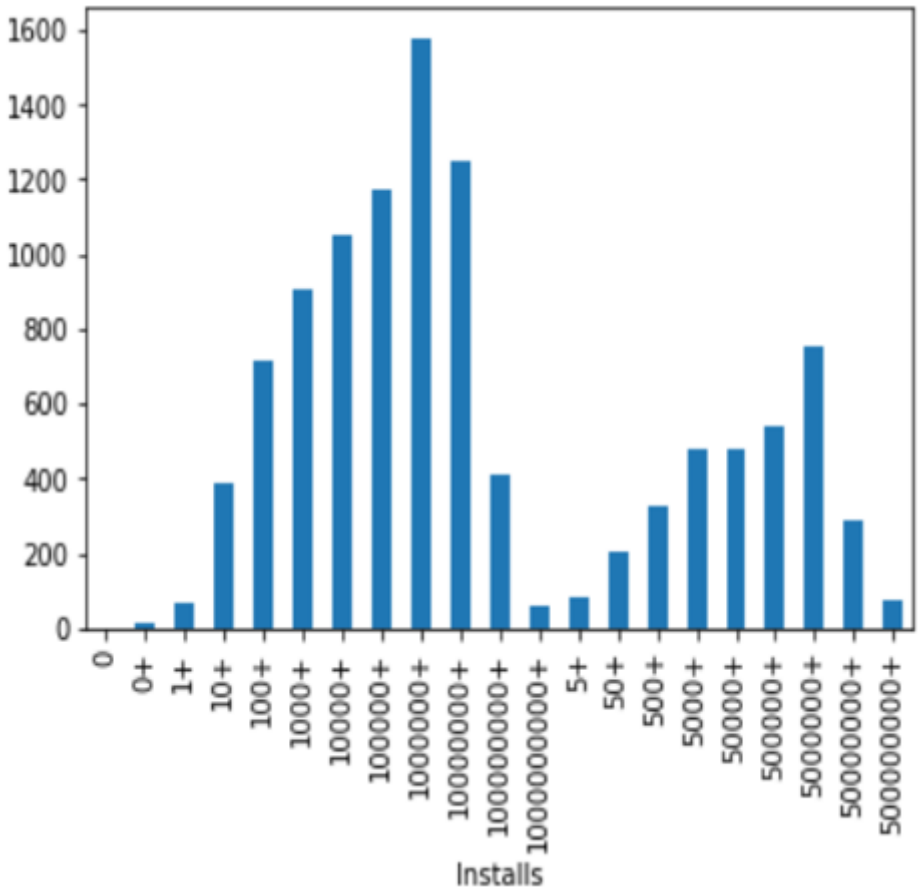

Figure 4

Count based on app's installed

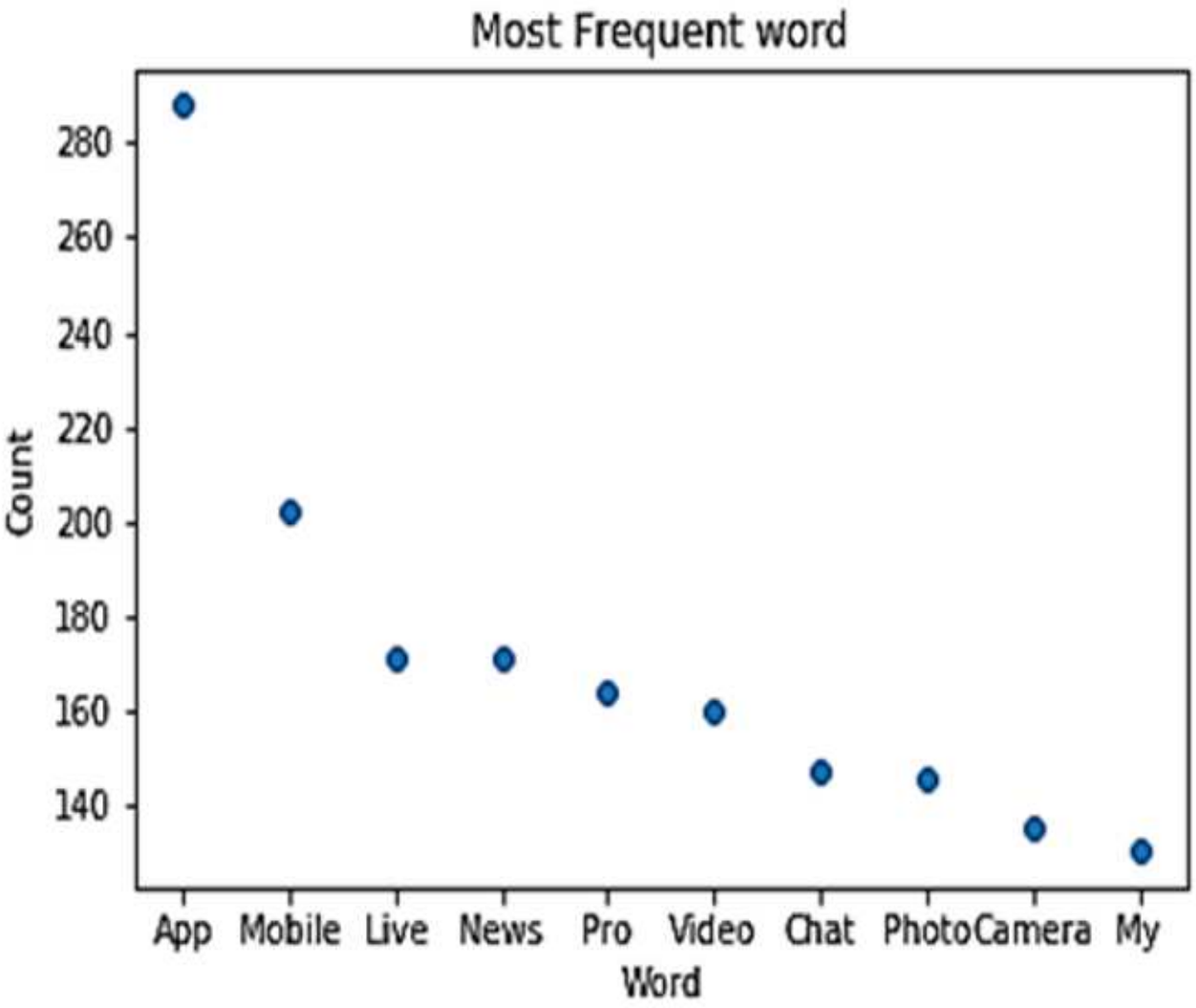

Figure 5

Most frequent word used in app name and its count based on usage

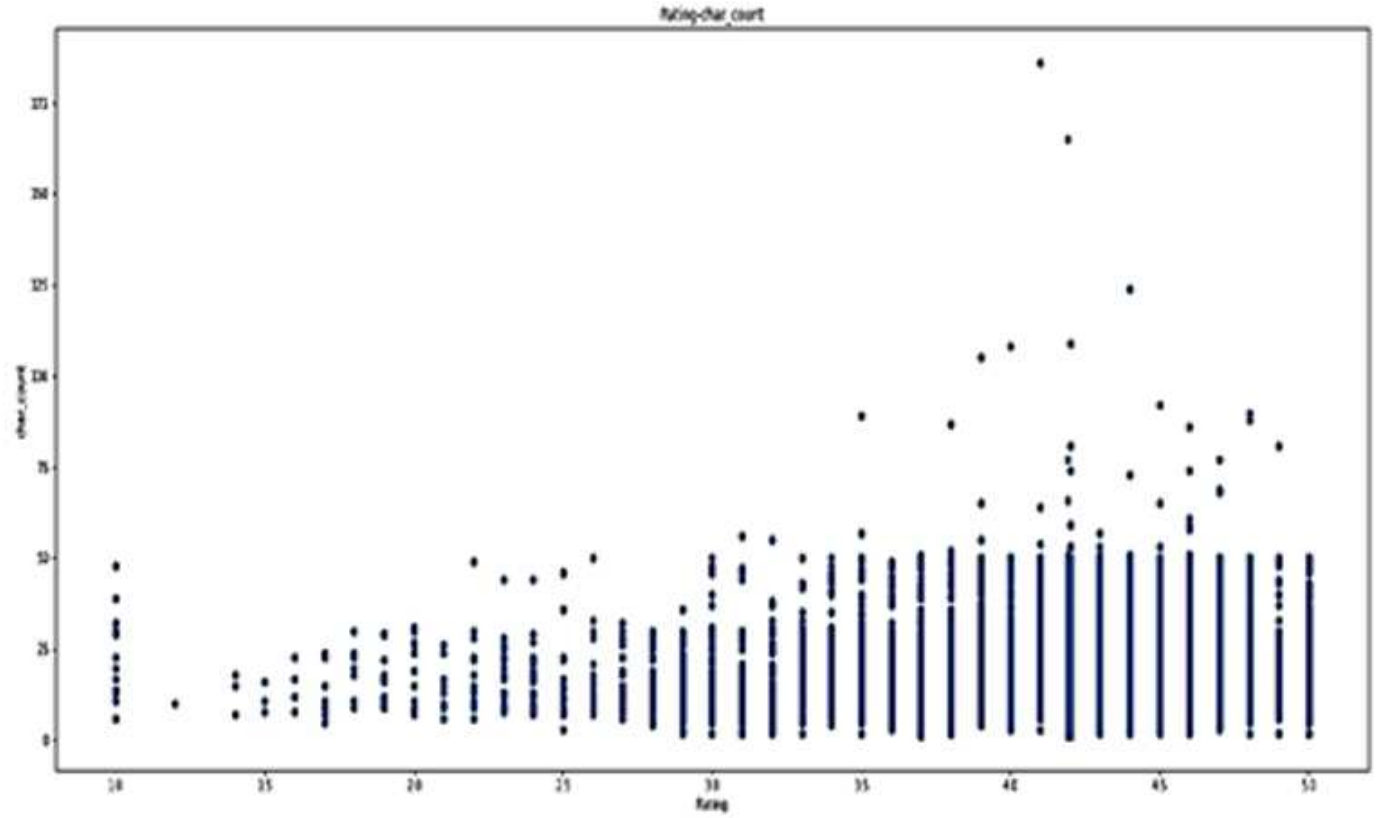

Figure 6

Ratings based on character count

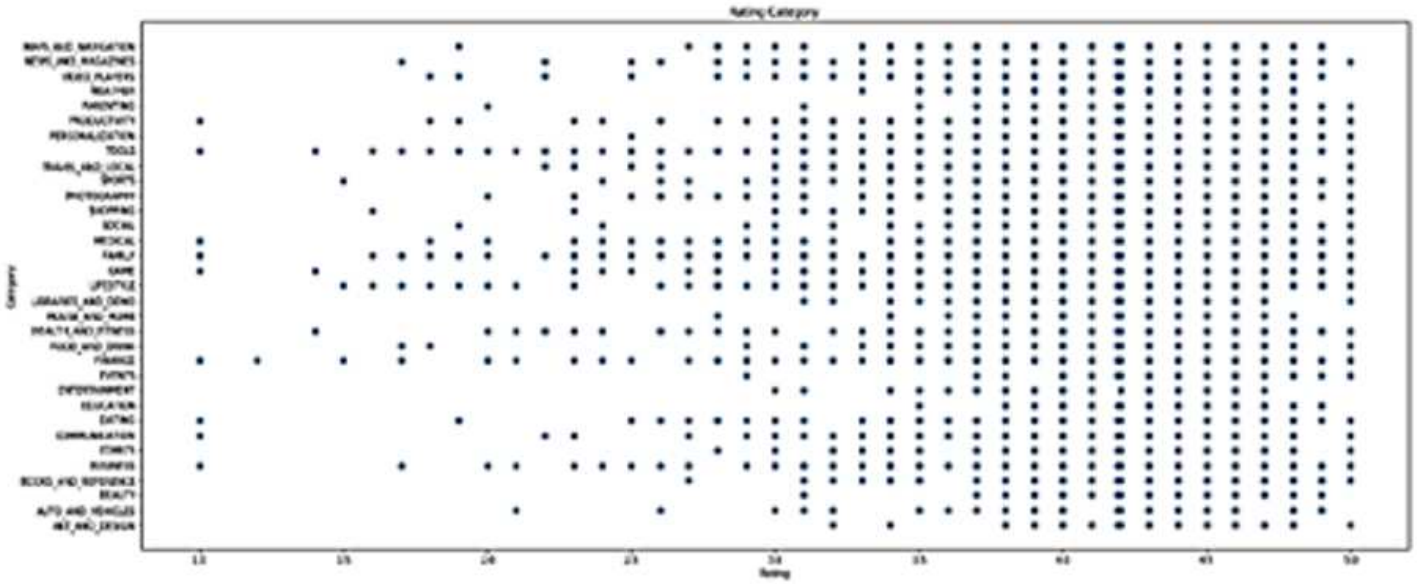

Figure 7

Ratings to different categories of app

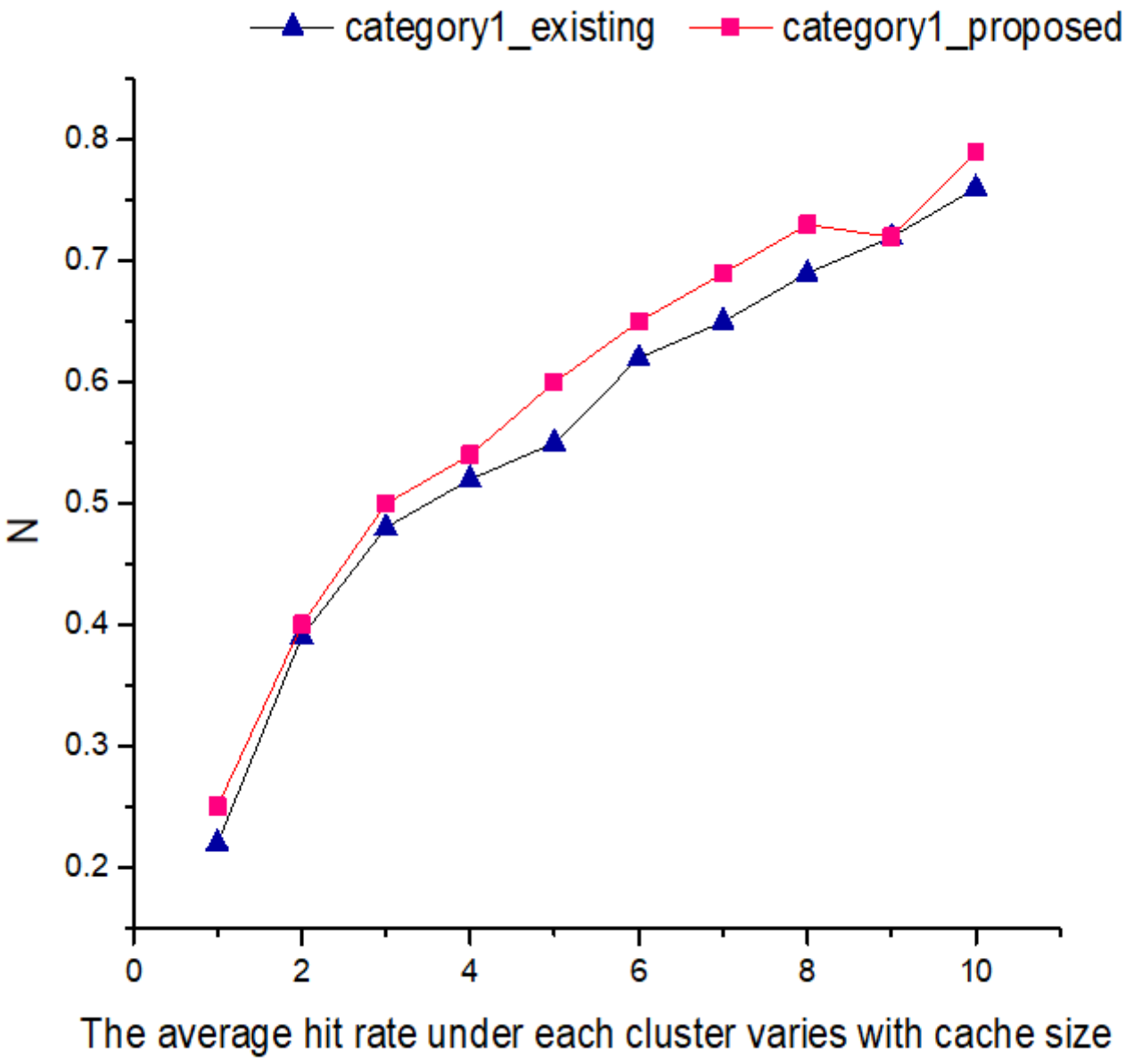

Figure 8

Comparison of 1st category of proposed and existing

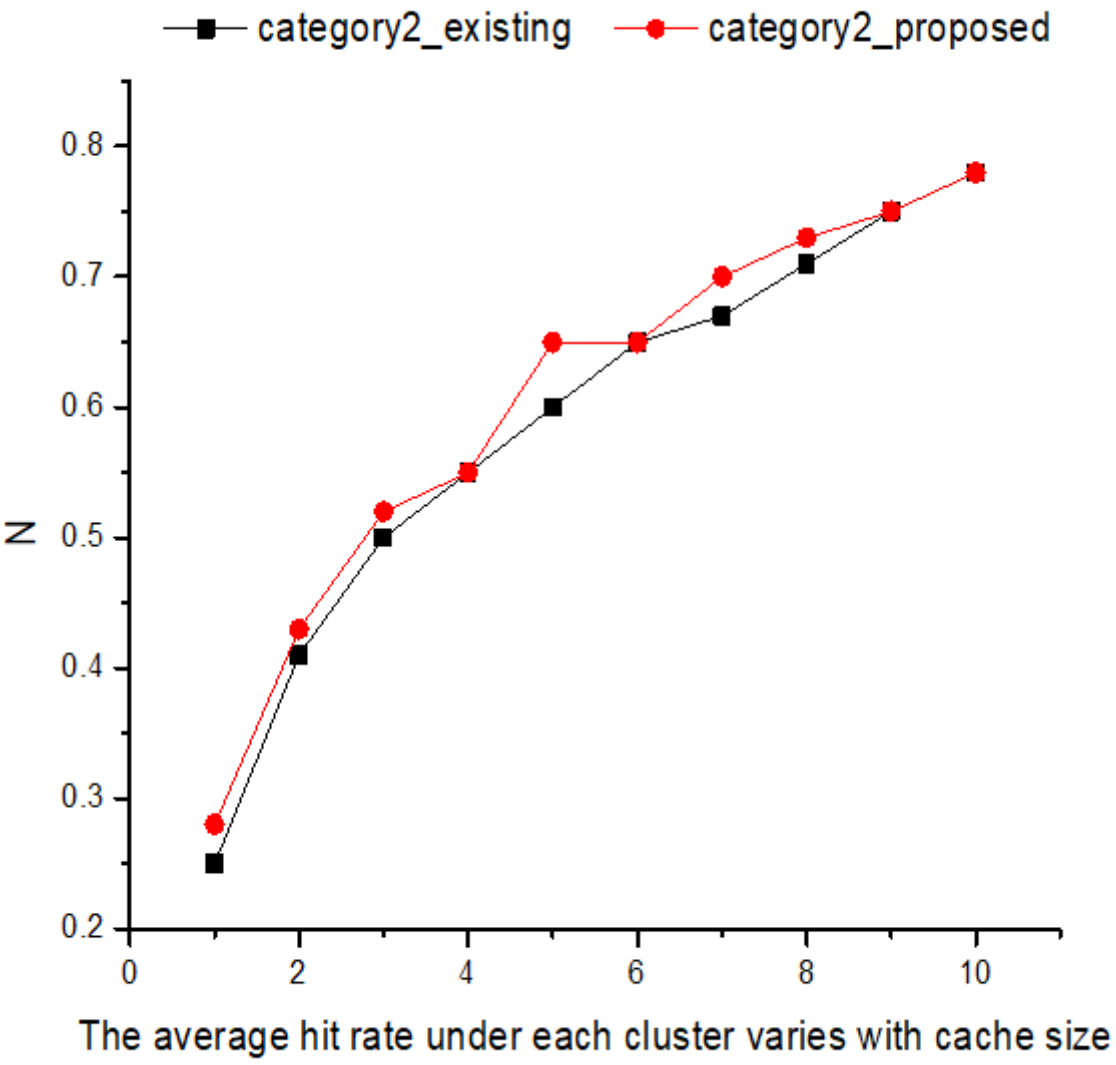

Figure 9

Comparison of 2nd category of proposed and existing

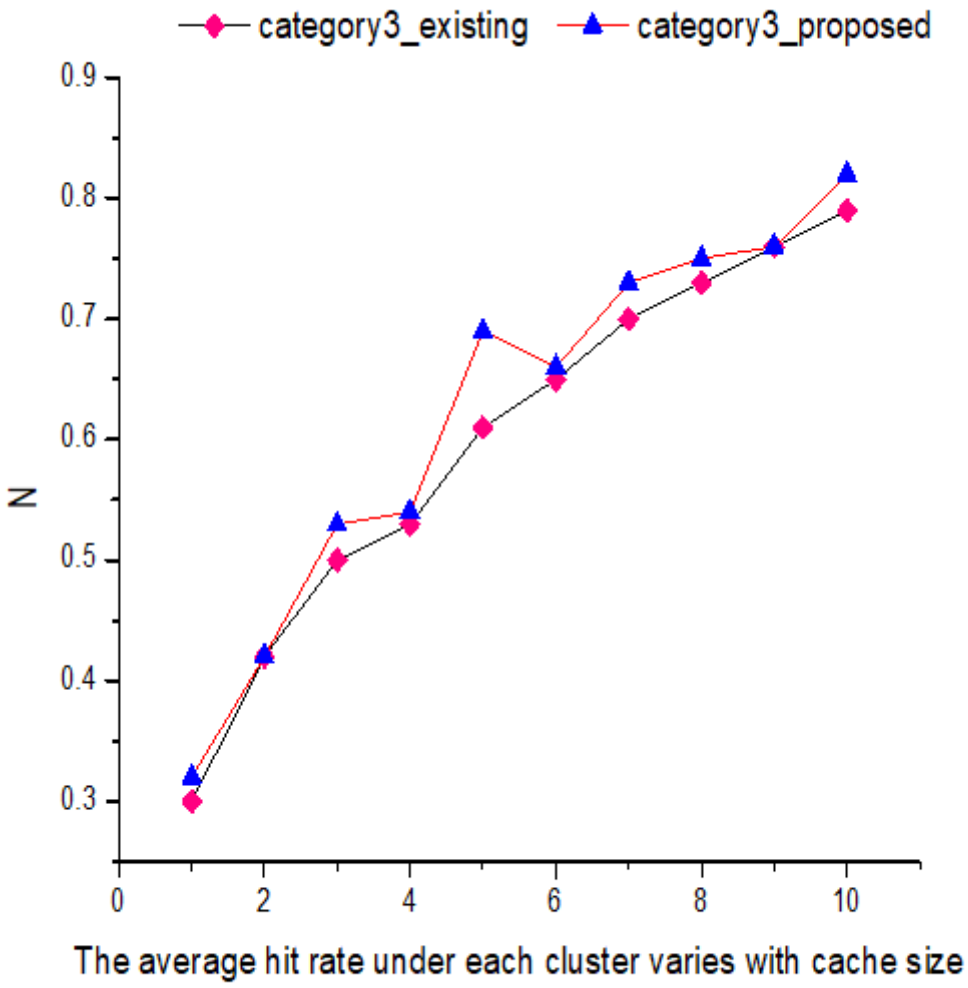

Figure 10

Comparison of 3rd category of proposed and existing

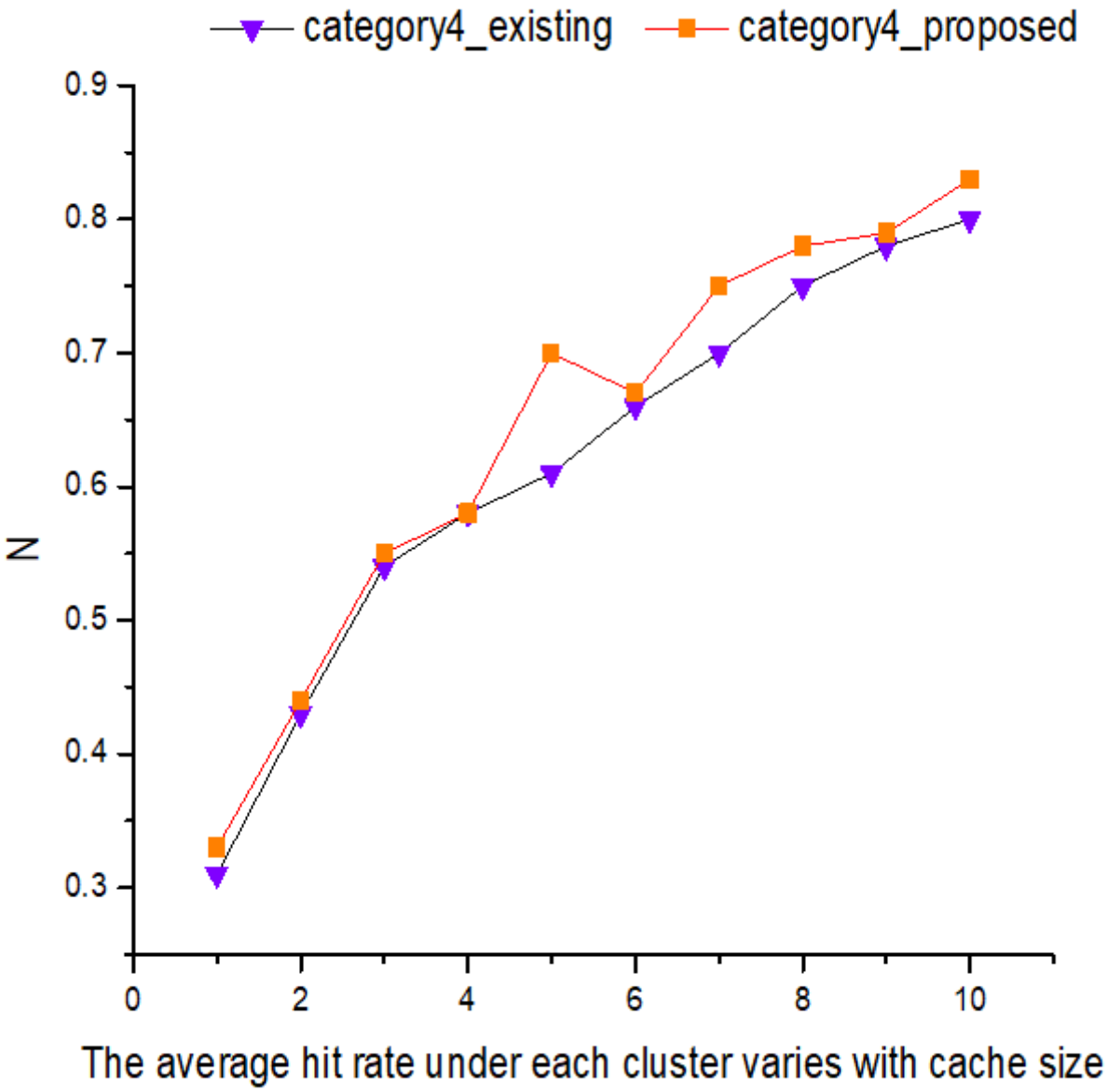

Figure 11

Comparison of 4th category of proposed and existing

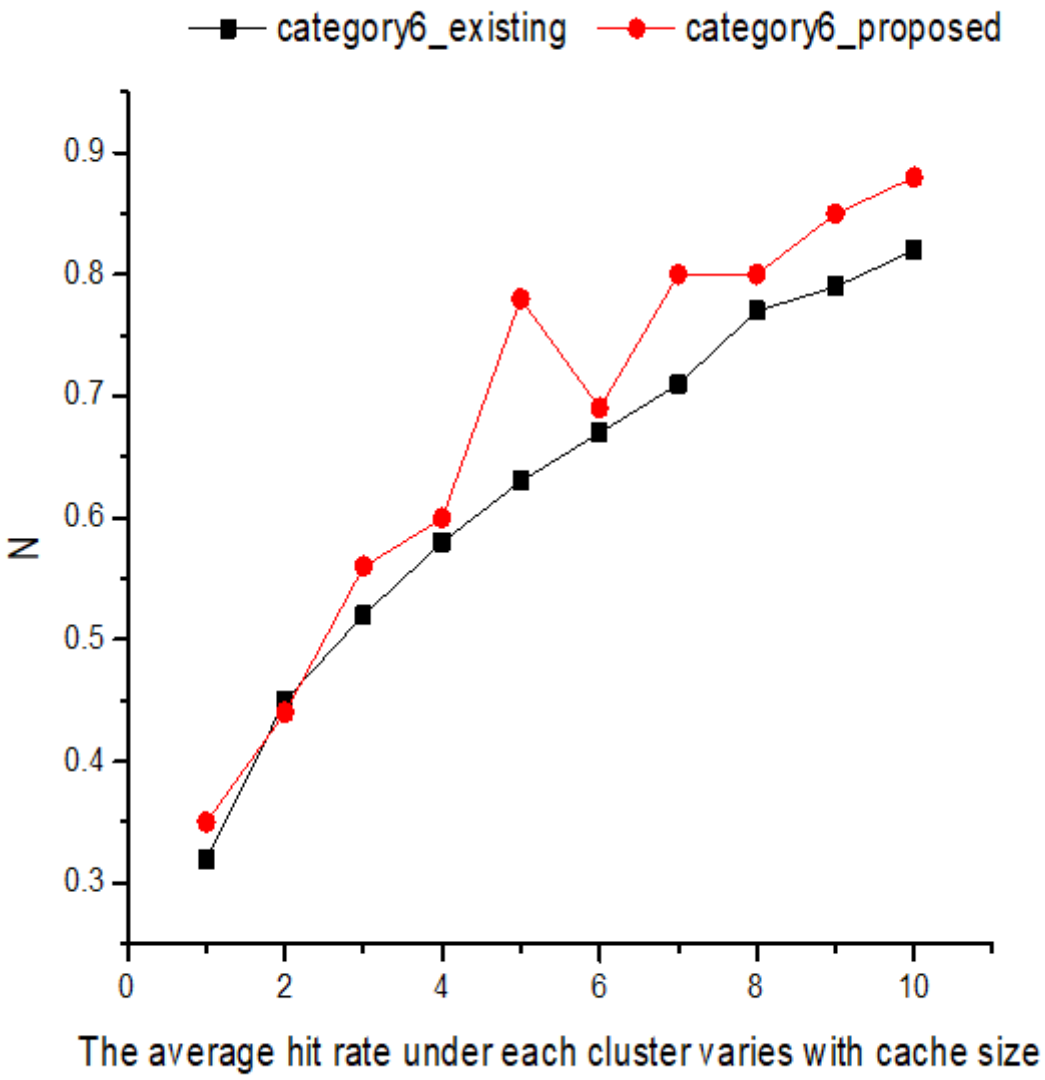

Figure 12

Fifth category comparison

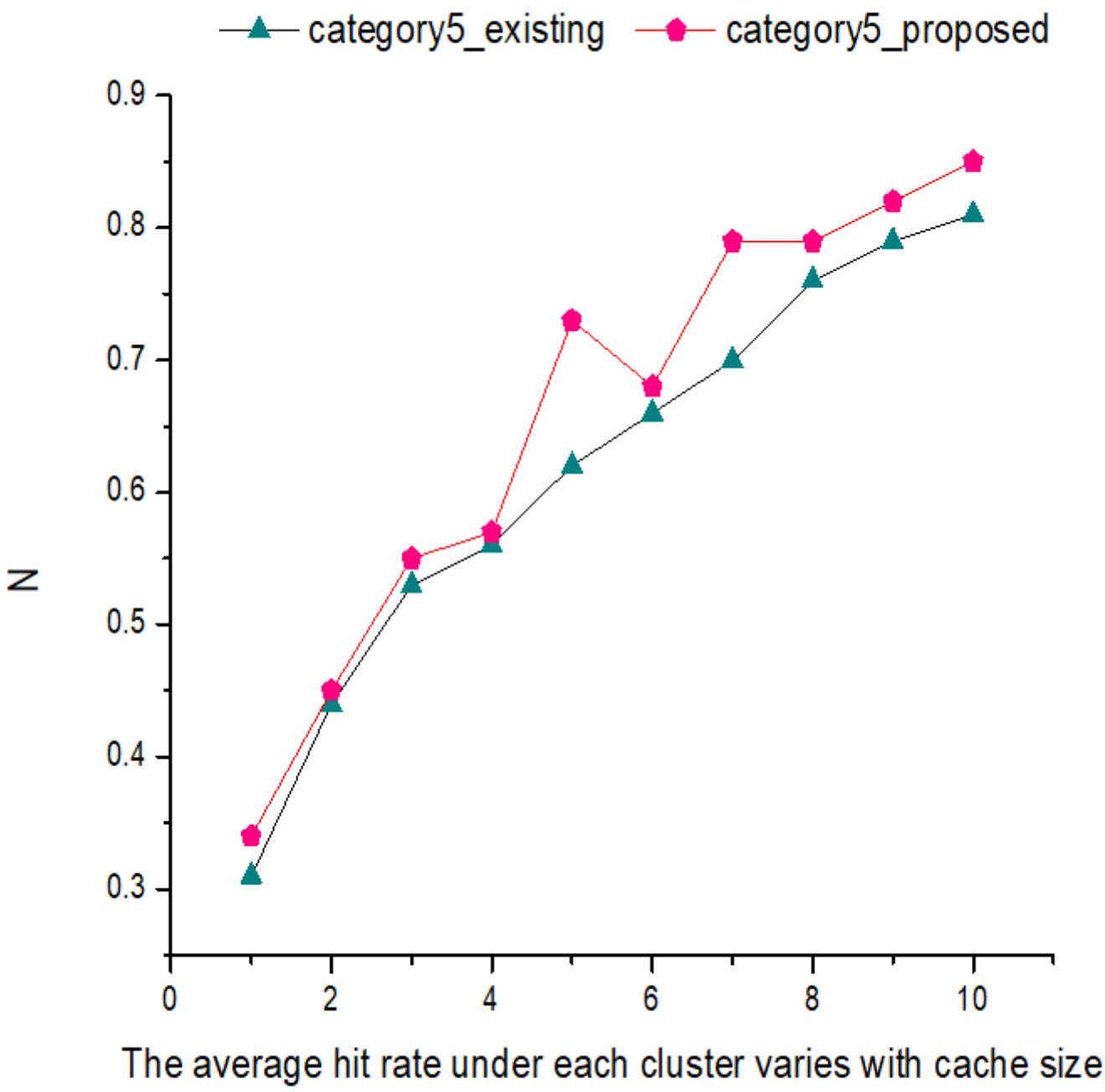

Figure 13

sixth category comparison