Krishnamoorthy PARTHIBAN, PhD Candidate
E-mail: parthibanresearch17@gmail.com
Associate professor Sundaram SUJATHA, PhD
E-mail: sujathasuce@yahoo.com
University College of Engineering, BIT campus
Anna University Trichy, Tamil Nadu, India

SIMILARITY-BASED CLUSTERING AND SECURITY ASSURANCE MODEL FOR BIG DATA PROCESSING IN CLOUD ENVIRONMENT

Abstract. *Present internet users evidently register the adoption of services offered through web resources. Presently prevailing scenario of adopting a cloud environment assists this practice in an intensified manner. Consequently, this influences the fabrication of an expertise web service recommendation system that serves with an ability to reach an utmost proficiency in retrieving web services. A prevailing search engine usually opts for providing recommendations for the query being posed by the user through utilization of a prevalent Collaborative Filtering methodology. In case of a accomplishing a web service from a cloud paradigm, the intricacy arises in processing the vast amount of information within a stipulated amount of time. However, the utmost challenge lies with the resource provisioning for those tasks demanded by user. In addition, effectual performance of the entire system gets degraded out of furnishing request in view of precedent user rating alone in an implicit way. In order to surmount over the issue of accomplishing an unrelated web service from a search engine, an inventive approach devised as Distance based Agglomerative Clustering and Secure Service Recommendation (DAC-SSR). A robust form of subset formulation via user-item and item-item similarity promotes the existence of resourceful associative similarity in between that user query and web service. Preprocessed dataset fed to the consecutive models comprises of various domains and proceed with the distance based agglomerative clustering. Those clusters get subjected to adopt with Similarity-ranking (S-r) scheming that tends to rank those clusters relying upon the similarity assessed. However, a robust user-item similarity also gets computed through the implication of a cosine similarity. Afterward, these ranked datasets acquire its position in a server as encrypted data by means of realizing Elgamal cryptosystem. Realizing efficacy of DAC-SSR gets accomplished through a comparative analysis of an existing methodology OGRPL-FW in terms of RMSE, MAE and running time metrics.*

Index Terms**:** *Agglomerative Clustering; Big data; Cloud computing; ElGamal encryption; Security; Web Service Recommendation.*
JEL Classification: **C00, C8**

DOI: 10.24818/18423264/52.2.18.11

INTRODUCTION

Incessantly surging intention of internet users to surf within a huge amount of information within a cloud environment in order to probe for an optimal service certainly necessitated the establishment of a web service recommendation system. Queries in search of some web services are postulated by the user constitute some sorts of patterns that are assessable for acquiring an abruptly pertinent web services with an ease[1]. This constraint is sufficed by a search engine by means of providing a lot of web services prevailing in the cloud based environment. Owing to existence of a huge lot of web services, the complexity arises with opting for a most suitable one[2]. Since there exists an extremely complicated circumstance regarding pertinent service retrieval, a prevalent approach that is capable of fulfilling the need of the hour is utilized termed as Collaborative Filtering (CF) based recommendation system, which positively deployed the content based procedures. This approach is skilled enough to characterize the services preferred by the user relying upon the past user fondness being archived. It simply works on a static supposition that the users be liable to opt for a similar services all through their way [3] which is practically infeasible.

Consequently, this sort of offering recommendation leads to a scanty recommender system that is inefficient to provide pertinent set of web services for the query being posed by the user into a search engine[4] that is certainly deployed in a cloud environment. Facilitation of a ubiquitous recommender system that tends to serve user on the basis of a Service Oriented Architecture (SOA) is highly necessitated to cope up with this fast growing technical scenario[5]. The recommender system deployed within a cloud based environment is thoughtfully forced to possess all sorts of QoS characterizations that accounts for a minimal response time, incurring a feasible cost, reliability and service availability. Migrating towards cloud computing also prompts for influencing some challenges in a real-time scenario that involves in dealing with a big data stream[6]. Challenges in a big data phenomenon arise in terms of 3V's defined as "Variety" that accounts for information gathered from varied set of domains such as sports, hardware and space research etc., "Volume" which states the constantly rising quantity of data and "Velocity" delineates the rate at which the data is being retrieved from the hefty information available[7].Beyond all these discrepancies, the security criterion matters a lot in adopting a web service [8]engraved in big data stratum. The location identification involved in adopting a web service acknowledges the whereabouts and activities of user.

The ultimate norm to assess relevancy between similar services or any sort of item being requested through user query is possibly abstracted through a proficient clustering methodology by means of utilizing rating information projected by the user[9]. Associations between the user request and the persistent knowledge possessed by the recommender systems regarding certain domains credibly enhances its efficiency[10]. Owing to the information overload convoluted in the recommender

DOI: 10.24818/18423264/52.2.18.11

systems deployed in a cloud based environment, the information is processed after subjecting to a robust clustering technique that probably frames a typical user-item and item-item subsets. Hence, a robust similarity assessment is mandated to furthermore rank the data unconcerned to the user ratings[11]. Those ranks provisioned by the recommender systems are prone to the security breach. Since, varied set of domains are not deliberated, uncertainty occurs in opting for a web service[12]. In order to overcome these issues, an effectual system that is capable of handling user queries with a proficiency accompanied by security measures is proposed by means of employing DAC-SSR methodology. The paper is organized as follows: The detailed description of the related works on recommending web services with privacy preservation is discussed in section II. The implementation process of Distance-based Agglomerative Clustering (DAC) and Secure Service Recommendation (SSR) is described in section III. The comparative analysis of proposed approach with existing methods provided in section IV. Finally, the conclusions about the application of DAC-SSR on the input data is presented in section V.

1. RELATED WORK

This section discusses the related works on web service recommendation system with the security criterion through the proficient subset formulation. Plentiful development in software utilization and online services escalated the issue of handling materials regarding suggestion making with web services and opt for better management applications with an adequate consumer affiliation.

Gani, et al. [13] investigated on varied feasibilities available with cloud computing paradigm and its intricacies exposed in managing the information prevailing in big data scenario based indexing systems. Owing to an immense progress of information accomplishing an enhanced throughput associated with a proficient data lookup was evidently complicated. the Kang, et al. [14] integrated QoS preferences and diversity feature interest on web services for recommendation system. By exploring the web service usage history, the user's interest and QoS preferences and candidate's score regarding web services were computed on the basis of potential user interest with QoS utility. Based on functional similarity between web services it was ranked and the web service graph is constructed with respect to scores derived along diversity degree .Chen, et al.[15] presented QoS-aware Web service recommendation approach to recommend the best predicted Web service QoS values on the basis of historical Web service QoS records for active users. Further, this QoS based recommendation system was required to improve the results in terms of scalability, accuracy due to unavailability of similarity computation. Then, the malicious users were detected with inaccurate QoS properties.

DOI: 10.24818/18423264/52.2.18.11

Wang, et al. [16]measured the reputation of web services by computing the feedback similarity in malicious feedback rating detection approach to improve the service recommendation process. After identifying the IP address with the offending feedback ratings, these ratings were blocked by using a standard Bloom filter. When the intensity of malicious feedback rating was low, then the rating detection approach caused failure. Also, the adjustment scheme of this measurement was unsuitable for a new service. Due to an availability of different feedback rating, the performance of service changed their characteristics suddenly. Hoffmann and Söllner[17] contemplated a trust-based recommendation system termed as Trust Supporting Design Elements (TSDE), constructed on the basis of the behavioral trust theory in order to eradicate the uncertainties that arose with users. Manipulated TSDE was capable of generating short-term trust on applications with respect to the security norm involved. Eliminating uncertain users were accomplished by associating antecedents and hence, required a surplus number of antecedents in tracing user behavior for resolving the instability in identifying the uncertainty convoluted in recommending a trustful behavior.

Sun [18]suggested normal recovery collaborative filtering approach (NRCF) address the problem of personalized web service recommendation. By investigating the characteristics of web service QoS values, the information of similar users was combined with information on similar web services to improve the prediction accuracy. Due to highly dynamic nature of internet environment, the QoS value of web services was changed with respect to time. This NRCF algorithm was unable to design in an online version and unsuitable for other application domains such as cloud computing. From these reviews, it is analyzed that the approaches mainly used for recommendation system are collaborative filtering, context-based approach, and hybrid approach. Wang, et al.[19]established an inventive friend recommendation system for people enrolled in social media through a semantic approach unlike deliberating social graphs for exposing like recommendations. The semantic approach trailed here usually abstracted the user preference on the basis of their habituated routine. User-centric sensor information that was composed of sensitive smart phones were assimilated to incorporate the user preference. The likeness between each users' life styles was assessed through deployment of Latent Dirichlet Allocation algorithm for defining a friend-matching graph. The endorsed system typically resembled the inclinations of user fondness in picking friends.

Yao, et al.[20]developed hybrid method of collaborative filtering and content-based recommendation in a three-way aspect model for web service recommendation. Simultaneously, the similarities of user ratings and semantic content of Web services were considered in a hybrid approach. But this system was unsuitable for personalized Web service recommendation and service clustering to appropriately use the content.Li, et al.[21] implemented a two-fold approach for resolving the issue of cold-

DOI: 10.24818/18423264/52.2.18.11

start by means of opting for gathering neighborhood data in an item-basis. The confidentiality and security criterion of the users were highly enhanced through a specific prototype constructed on the basis of differential privacy. The procedure of Collaborative Filtering (CF) was assisted with varied shared data algorithms framed with an integration of neighborhood enhancing procedures. It accomplished an enhanced preciseness for recommendation system but incapable of deliberating the privacy for users on the whole.

Vera-del-Campo, et al.[22] tested a DocCloud prototype that intended in safeguarding all sorts of users from legitimate attacks. In specific, the prototype examined about the properties possessed by recommenders such as anonymity and a plausible deniability. It also operated on stating a legal responsibility of both sorts of a cloud as well as the virtual machine dwelling within a cloud. Hence, conveyed a robust system for proposing recommendations to the customers at one go and those were also enabled with a deniability in a plausible manner. The capability of the recommendations delivered by the system completely relies upon the suppositions provided by users and the quality is left uncertain.

Hoens , et al.[23] fabricated a privacy-friendly structure through architectures differed in twofold ways such as Anonymous Contributions Architecture (ACA) and Secure Processing Architecture (SPA). Multi-party computation methodology was trailed in SPA for ensuing recommendations for the patients enrolled. The ratings offered by the patients were done in a protected forum which maintained the personal data in a concealed mode. The associating link was left unconcerned between the patient and the recommender system in ACA. It provided an enhanced security accompanied with reliability on preserving patient detail while incurring an extraordinary computational load owing to lack of optimized functionality in providing recommendations.

Toch[24]introduced a framework formally termed as Super-Ego in order to preserve the data associated with the position and was made feasible through a crowdsourcing structure. This Super-Ego framework was capable of envisaging the user's choices regarding privacy criterion with respect to the population available. The limitations inferred with the devised framework was,

- Impending privacy risks initiated with sharing of location were left unsettled
- Bounded applicability owing to single filtering approach
- Absence of concealing confidential locations led to information abuse

Dandekar, et al.[25]evaluated an auction procedure for exploring the assurance of privacy for information owned by every individual. Visibly identified weighting functionality was assigned to the linear predictor that involved in the plotting of statistics for information possessed. A feasible trade-off was realized by means of implementing Discrete Canonical Laplace Estimator Functions (DCLEF).

DOI: 10.24818/18423264/52.2.18.11

[26]employed an effectual web service clustering methodology termed as Web Service Tag Recommendation (WSTRec) approach through an incorporation of tags as well as Web Service Description Language (WSDL) documents. The confinement found with a presence of the noisy tags as well as issue of irregular tag dissemination was alleviated through WSTRec. Skillful recommendations were made out of deploying approaches like assessing the semantic relevancy of tags, its co-occurrence and tag mining. WSTRec exhibited a comparatively better outcome than conventional web service clustering approaches centered on WSDL. However, the capability was mitigated to a highly confined bound.

Hu, et al.[27]examined the diverse set of methodologies that was capable of fusing information drained out of user suggestions as well as social media. Initial methodology accounted for deployment of a graphical neighborhood framework through spreading out Social Matrix Factorization (MF). An inventive prototype that collated both the aspects informed in prior along with the item reviews termed as Model of Rating, Review and Relation (MR3) was devised to rate the perfectly lined up hidden topics and the concealed factors. The proficient prediction was accomplished with deliberated model along with some sorts of added restrictions. Unsuitable suppositions regarding hidden topics as well as latent factors. Incorporation of inherent feedback was left unnoticed and hence, observed for degraded performance. Bu, et al.[28] formulated an innovative Multiclass co-Clustering (MCoC) approach for deriving the precise associations between user-to-user and user-to-item in a simultaneous fashion. Top-N number of recommendations were derived by means of collating subgroups created and the conventional Collaborative Filtering (CF) algorithms. This lacks in optimal subgroup utilization out of manifold subgroups being framed.

Zhao, et al.[29] endorsed an online social recommendation structure by appending both Online Graph Regularized user Preference Learning (OGRPL) together with the Frank-Wolfe (FW) procedure. This was capable of resolving the issue of optimization in an online community while consuming a lengthened amount of running time in terms of processing the online data retrieval with an utmost relevancy. Likewise, the error values such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) were produced with the mitigated performance and hence, found escalated with prevalent Hybrid Recommendation (HR) [20]approaches. This section reviewed the traditional methodologies and techniques prevailing for providing recommendations online on the basis of similarity and ranking produced for web services pertaining to user reviews gathered. Out of those lucrative tendencies exhibited by the prevalent approaches, the intricacies acquired are,

- Predicting a finest user-item subgroup accounts for a critical issue and optimal formulation of subgroup is unfeasible
- Opting to preserve the user information deliberating their privacy preference

DOI: 10.24818/18423264/52.2.18.11

- Prevention of unauthorized access in prior to provision of recommendation is a huge lacking criterion
- Facilitation of uncertain or an irrelevant recommendation in terms of similarity is devastating
- Affording ample security centered on the basis of position based secure environment is certainly vague

In order to overcome these issues, an inventive proposal is accomplished by generating an expertized system that is capable of handling user information enabled with privacy preserving criterion along with facilitation of acquiring pertinent web service recommendation.

2. DISTANCE-BASED AGGLOMERATIVE CLUSTERING (DAC) AND SECURE SERVICE RECOMMENDATION (SSR) FOR RECOMMENDING WEB SERVICES

This section discusses the implementation details of Distance-based Agglomerative Clustering (DAC) and Secure Service Recommendation (SSR) for recommending web services. Fig. 1 shows the workflow of DAC-SSR to reduce the security breach and to escalate the relevancy of delivered web services through clusters formed out of applying agglomerative clustering methodology. Initially, the rows with unfilled entries are removed and distinct keywords are abstracted through preprocessing. The dataset contains the text documents that comprise of several domains regarding sports, hardware, space, politics, religion etc. The workflow includes the major processes listed as follows:

1. Big data processing
2. Preprocessing
3. Agglomerative clustering devised on Distance assessment
4. Similarity-rank (S-r) scheming
5. ElGamal Cryptographic Structure(ECS)

DOI: 10.24818/18423264/52.2.18.11

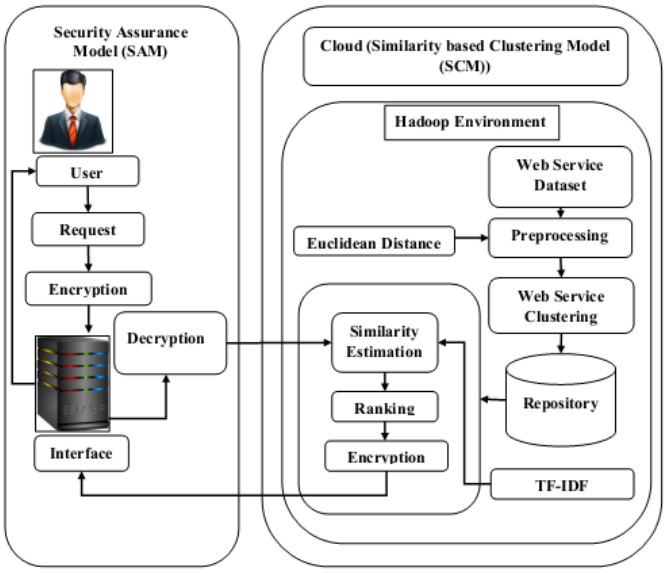

Figure 1. Work flow of proposed DAC-SSR

The formulated architecture is completely simulated in the cloud environment. Owing to the presence of the huge amount of data in the cloud, this DAC-SSR prototype is realized within a Hadoop MapReduce framework in order to handle the big data with an enhanced proficiency. As aforementioned, the web services utilized are engraved within textual documents and are preprocessed to acquire the isolated root words. Initially, Similarity based Clustering Model (SCM) tends to act into the unprocessed datasets. This preprocessing methodology assures the mitigation of complexity involved retrieving pertinent documents that account for embracing highly associated web service. Furthermore, these preprocessed documents are implied to get subjected to undergo a procedure of agglomerative clustering. The Euclidean distance gauged between clusters are deliberated for clustering those keywords regarding web services by means of deploying a bottom-up approach. Afterward, the clusters are evaluated for a similarity measure in between them. It is scaled on the basis of semantic vicinity among the clusters. Highly analogous clusters are provided with an utmost rank and it decreases linearly with the depletion of similarity between them. These ranked web service documents are archived on the web server after encryption. At this juncture, Security Assurance Model (SAM) comes into action in order to maintain the provision of recommendations in a secure manner. ElGamal cryptographic methodology assists in prompting for retrieval secure web service through the provision of a common public key on both client side and server side. Moving on to the side of the user, request projected is assessed and encrypted before giving into the server for further

DOI: 10.24818/18423264/52.2.18.11

processing. Moreover, the user query is decrypted and expedites in search for a pertinent web service. The web service quested by user query concerned is offered with appropriate one after decrypting it in prior to reach the user. Thus, a recommendation system enabled with security criterion is articulated.

2.1 Big data Processing

Incorporating a large set of web services from diverse group of domain unquestionably imposes the need for accomplishing a cloud based procedure owing to manage the memory constraints in accordance to the restriction laid with respect to time incurred for processing. In order to accomplish a proficient paradigm in accomplishing web services archived within the repository placed in a cloud environment is achieved via employment of this big data processing procedure. The cloud computing environment encompasses the both web service repository as well as similarity appropriating mechanisms. These entire setup is placed within the Hadoop architecture that is obviously placed in a cloud platform. In order to optimize the overall operating time span for processing vast amount of information available within cloud, the Hadoop framework is utilized. Several user queries are possibly processed at single instance of time in a parallel manner within Hadoop framework. These preprocessed web services deposited within the repository dwelling in a cloud environment is checked for its relevancy with respect to the projected user query at the time of web service retrieval.
The parallel execution of user queries are made feasible through incorporation of a MapReduce framework [30]. It is actually carried out in a two-fold manner stated as Mapper and reduce phase. Initially, the map phase tend to grasp dataset as input from where it is previously stored. The perceived data are mapped as per the instruction prescribed by user query and hence, the data gathered are mapped into separate functional procedures. Afterward, these mapped data are shuffled and allocated at every instant of time in a simultaneous manner to the several reducers available respectively.This reduce functionality typically holds the responsibility for mitigating the overall timespan stipulated for processing.

2.2 Preprocessing

The given dataset encompasses some sort of unoccupied rows or existence of annoying entries. In such a case, those instances are handled through varied processes involved within the preprocessing. Each and every text manuscript inscribed in the dataset is scanned as per the user request specified. The procedure of preprocessing usually checks for the all sorts of entries prevailing within the documents in a manner that each and every character is crosschecked. The preprocessing procedure trails a standard way of excavating distinctive words by reading through alphabetic characters subsisting within documents that are delineated by blank spaces and all other character

DOI: 10.24818/18423264/52.2.18.11

series are skipped over except wholesome words that sound meaningful. Those words available within the document are fragmented into manifold parts formally termed as tokens through a process of tokenization. Entire character stream indulged in a document are parted by means of utilizing delimiters like space and punctuation marks. Let the complete set of characters involved in the textual documents of the dataset utilized be incorporated within the linear manifestation.

$$\omega M\langle A^{"}\rangle AM\omega$$

For the complete document being processed ωdefines the blank symbol existing,"signifies the apostrophe, Other punctuating marks are defined through$M$={.|,|;|:|?|!|"|′} and finally, all sorts of alphabetic characters are recognized through a standard set defined as$A$={$a|b|\ldots|z$}$^{+}$. The content prevailing in the document is appropriated to follow a single language and it is deliberated as a basic supposition for the complete procedure tracked. Hence, a typical learning methodology is capable of managing these documents loaded.

Stop Words Removal

The stop words alleviation from the textual documents of a dataset is accomplished through a standard approach implied on the basis of a dictionary. The algorithm is employed for getting rid of the existing stop words is employed by,

*Step 1:*Utilize an individual array to archive the tokenized document concerned

*Step 2:*Stop word list is preserved and those entries are verified with the validating target document

*Step 3:*A progressive search methodology is followed to endorse the textual word in the document with that of list in a comparative manner

*Step 4:*On realizing a counterpart between those words equated, that particular word is eradicated and the search strategy extends till the array ends

*Step 5:*Reiterate the similar procedure from step 2 to step 4 until all sorts of stop words are cross verified with words in that document

*Step 6:*As a final point, the document isolated out of stop words is acquired along with indicators that designates the total counts of stop words detached from the document examined, overall count of words available in the marked document after processing, total number of words resourcefully available within the document and number of discrete stop words originated from the document.

Stemming

The document freed from stop words is furthermore processed to liberalize from the varied syllabic configuration convoluted in the key terms that usually inhibits the ease of search. The process of altering the word to be bear a resemblance to its root form accounts for stemming. On exterminating those morphed form of syllables, exact key terms are acquired and consequently, the reclaiming efficacy is abruptly enriched along with mitigation in overall dimensions of the indexing files. The standard constraint deployed for predicting stemmers emulates a varied set of techniques such

DOI: 10.24818/18423264/52.2.18.11

as n-gram stemmers, successor variety, table lookup methodology and Affix removal stemming. Here, in this paper, the methodology for detaching stemmers by means of recognizing surplus suffixes existing in the core words is followed. Strategy applicable for accomplishing stemming procedure pursues a typical set of protocols.

Case 1: On realizing a word with ending "ies" but not with "eies" or "aies"
then Replace "ies" with "y"

Case 2:On realizing a word with ending "es" but not with "aes" or "ees" or "oes"
then Replace "es" with "e"

Case 3: On realizing a word with ending "s" but not with "us" or "ss"
then Replace "s" with "NULL"

After completion of these procedures, the textual document being processed contains only the distinctive keywords (*Dk*) that are completely resourceful in nature.

2.3 Agglomerative Clustering devised on assessing Euclidean Distance

The web services contained within the preprocessed datasets that encompass only root words are further clustered through a hierarchical clustering policy termed as agglomerative clustering. Here, the preprocessed datasets are assessed for a distance prevailing in between them and are segregated into fragments in a progressive manner. The fragments acquired are subjected to frame layers of the nested structure. It is accomplished by means of establishing a tree structure by gathering the objects organized into varied layers on the basis of distance realized. This methodology certainly alleviates the necessity of computing total number of clusters prevailing in. Hence, it is decidedly apt for getting applicable in a cloud environment that comprises of a large amount of information engraved on it. The process of clustering trails a formal procedure for indicating the stoppage of clustering mechanism. It is indicated as,

While:
It is an unprompted stage to stop clustering;
do
Pick two clusters dwelling within a minimal range of distance;
Assimilate those picked ones into a single cluster;
End

Distance assessments between clusters are measured in terms of vicinity or similarity metrics prevailing in between the data objects observed within the documents. Here, in this paper Euclidean distance is computed in order to cluster the inferred datasets that are preprocessed in prior. The Dk abstracted from the datasets are scrutinized for comparing with those datasets being organized. Those computed Dk are subjected to undergo the Term frequency (Tf) assessment in which, the total number of times a concerned word appears in each and every document prevailing in the dataset

DOI: 10.24818/18423264/52.2.18.11

are computed. *Tf* values assessed for all *Dk* abstracted from the preprocessed datasets are positioned within an array structure. The documents placed in successive columns that ranges from $1\,to\,n$ are represented by two different variables such as$q_1, \dots, q_i$and$p_1, \dots, p_j$. Hence, the Euclidean distance (Ed) is assessed between those two tuples existing in that particular array as,

$$Ed = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_i - p_j)^2} \quad (1)$$

The minimized amount ofEd obtained out of entire tuples is considered and those two are merged initially to form Merged data(Md). The similar procedure is trailed until the saturating point is reached by checking all *Dk*.

2.4 Similarity-rank Scheming

The recommendation of web services happens in a cloud-based environment by means of assessing the similarity between those data dwelling within merged data and decrypted request that is carried out in a dual-fold manner. Here in this paper, the similarity measure is computed in terms of item-item and user-item similarity. Initially, the similarity measure of item-to-item is defined through a *Tf –Idf* calculation. *Tf* as well as Inverse Document Frequency (*Idf*) is estimated to provide Relationship between the *Dk* and merged document clusters. For making an ease with representation, *Dk* is represented as m and *Md* is denoted by n. With the intention of working out term frequency attribute, the total number of times particular distinctive key term appears within a specific document is assessed.

$$f_{mn} = frequency\ of\ m\ in\ n \quad (2)$$

Moreover, the normalized form of term frequency is manipulated to abstract the overall frequency of the term existing within that cluster.

$$Tf_{mn} = {f_{mn}}\Big/{max(f_{mn})} \quad (3)$$

Now, the availability of *Dk* within a single cluster is identified in a crystal clear manner but this does not suffice for obtaining the overall item-to-item similarity measure. It is accomplished by means of collating both Tf and Idf measure. This certainly addresses the frequent existence of particular key term among the whole *Md*.

$$Tf_{mn}Idf_{mn} = Tf_{mn} \times \log\frac{N_m}{nf_m} \quad (4)$$

Here, term frequency resembled as Tf_{mn} of the each and every key term is incorporated with a ratio of aggregated clusters present within merged data to the frequency of the certain key term prevailing in the entire cluster collection. On assessing these measures, the similarity prevailing in between the distinctive key words abstracted and the clustered merged data is found on the basis of item-item

DOI: 10.24818/18423264/52.2.18.11

similarity. This aspect positively diminishes the overall time span consumed for procuring an appropriate document by probing through the clusters framed out of processing the textual documents that comprise of web services.

In the next step, the similarity between merged data and the Decrypted request(Dr) projected by the user which is archived within the server is assessed through an appropriation of cosine similarity values. The similarity between varied components is possibly identified only through this cosine similarity assessment. The value of cosine similarity usually ranges from -1 to 1.

Case 1: On obtaining the value '1'

It opts for accomplishing a decision, which states that the similarity between considered user request and the particular *Md* is inversely associated in the sense that if the frequency of the particular term searched for is surging up in the *Dr* then the document obtained is with the deflated frequency of the searched term for.

Case 2: On obtaining the value '1'

This value opts for a resolution that both user requests decrypted and the acquired cluster from merged data is exactly similar to one another

Case 3: On obtaining the value '0'

Attaining this value certainly, concludes the decorrelation prevailing in between the user request and the merged data. This orthogonality points toward the abrupt dissimilarity prevailing between the compared ones.

The variable(A_k)typically resembles the(Md) while(B_k) signifies the Dr and estimation of cosine similarity is given as,

$$Cs_{ij} = \frac{\sum_{k=1}^{n} A_k B_k}{\sqrt{\sum_{k=1}^{n} A_k^2}\sqrt{\sum_{k=1}^{n} B_k^2}} \quad (5)$$

With the above equation (5), the Cs assignment is converted into the S-r scheming functionality. Thus on incorporating both the similarity estimation functionality, the S-r scheming is accomplished with proficiency. The request projected by the user from the client side is penetrating into the server. After decryption, it procures for a pertinent web service prevailing within those preprocessed datasets. The measures such as *Tf-Idf* and Cs_{ij} tends to find the pertinent service by incurring an optimal time span. Opting for a better service with respect to a user request is diversely processed and hence, the overall processing time of attaining an associated web service is definitely reached. Afterward, the web service that possessed a high similarity is ranked accordingly with an utmost value while the service that possessed the least similarity is ranked with a smallest value. In order to avoid the mess up with the ranked procedures, these details are archived in the server after employing an

DOI: 10.24818/18423264/52.2.18.11

encrypting procedure. After optimizing the probe for acquiring a highly identical service, the processed web service is prompted for the requested user.

2.5 ElGamal Cryptographic Structure

A reputed cryptosystem utilized for creating a secure environment for accomplishing pertinent recommendations is ElGamal cryptographic structure (ECS). It is highly rigorous to penetrate into this structure by the intruders since it does not rely upon the conventional Public Key Infrastructure (PKI). This strategy is completely influenced by a Digital Signature Algorithm (DSA) generated on the basis of Diffie-Hellman key exchange procedure. Initially, the secret key formed is constructed relying upon the base point being generated from the Elliptic Curve Diffie-Hellman (ECDH) key exchange algorithm. Primarily, the base point is suggested from the Elliptic Curve that certainly suffices with the condition given as,

$$y^2 = x^3 + ax + b \mod p \quad (6)$$

The points abstracted from the elliptic curve is responsible for generating the proliferating assembly of integer modulo typically stated as prime p and primitive root modulo g. These integers are further utilized for generating the secret keys that hold the responsibility for crafting a common secret in between the user as well as the web server involved in. Those integer values fabricated are augmented with the data to be transferred from the admin to user and hence, the private key belonging to both user as well as admin is created. By means of utilizing this, a public key is formulated by both user and admin. Furthermore, the public key contrived is used to create the secret key in an individual manner. The public key generated by the admin is employed to manufacture the secret key possessed by the user and vice versa.

After devising the secret for both admins as well as the user, the process of encryption is carried out in a discrete manner. The request projected by a user to the web server is patched up with the secret produced by the user and promoted by means of augmenting the public key possessed by the user. Thus the user request is encrypted to form the Encrypted request (Er). On reaching the server, the Er perceived is again patched up with public key enunciated by admin. Afterward, the secret articulated by the admin is implied to decrypt the user request (Dr) and is further proceeded for processing. After fetching the pertinent web service requested by the user through the DAC-SSR procedure, the recommended web service that possesses an elevated rank value is encrypted by augmenting encrypted secret (Es) and is transferred to the user. On reaching the user it is decrypted by segregating the decryption secret (Ds) from that recommended web service.

Base point Generation in ECDH
Initialize a=11, b=22, u=29, x=3, y=1,s=0
$a = \left((x^3 + ax + b)\%u\right)$

DOI: 10.24818/18423264/52.2.18.11

if *a!=0* **then**
  for*i*=1: *u***do**
    $s = ((s + i)\%u)$
    if *a==s* **then**
      break
    else then
      $y = y + 1$
      $i = i + 2$
    end if
  end for
end if
if $y^2 = x^3 + ax + b$ **then**
$g = x$
$p = y$
end if
we got *g=3, p=11*

Secret key Generation in ECDH

a ← *private key (user)*
b ← *private key (admin)*
A ← *public key (user)*
B ← *public key (admin)*
S1 ← *secret key (user)*
S2 ← *secret key (admin)*
$A = ((g^a)\%p)$
$B = \left(\left(g^b\right)\%p\right)$
$s1 = ((B^a)\%p)$
$s2 = \left(\left(A^b\right)\%p\right)$

ElGamal Encryption for user

Pr ← *private key*
Pu ← *public key*
Es ← *encryption secret*
$h = (s1^{Pr})$
$Es = (h^{Pu})$
Sd ← *select data*

DOI: 10.24818/18423264/52.2.18.11

Er ← *encrypted request*
while*Sd.data*
$Er = Sd.data * Es$
end while

ElGamal Decryption for user
Pr ← *private key*
Pu ← *public key*
Ds ← *decryption secret*
$h = (s2^{Pu})$
$Ds = (h^{Pr})$
Dr ← *decrypted request*
while*Er.data*
$Dr = Er.data/Ds$
end while

ElGamal Encryption for admin
Ed← encrypted *data*
while*Rd.data*
$Ed = Rd.data * Es$
end while

ElGamal Decryption for admin
Dd ← *decrypted data*
while*Ed.data*
$Dd = Ed.data/Ds$
end while

At once when the data is opted and ranked in prior to the cloud storage, the security is undoubtedly assured. Preprocessing performed in prior to clustering abruptly mitigates complexity in probing a web service. Hence, attainment of a pertinent web service without any sorts of the security breach is an added advantage with the realization of Elgamal decrypting methodology accompanied with the ECDH approach. The pseudo code to implement the DAC-SSR for accomplishing pertinent web services in a secure manner is as follows:

DAC-SSR pseudo code
Input: Dataset (D)

DOI: 10.24818/18423264/52.2.18.11

Output: Encrypted Ranked data *Rd*

D←*loaddataset*
Do ← *domain*
Pd← *preprocessed dataset*
for*i=0*: *D***do**
  for*j=0*: *Do***do**
    Pd←*stopwords and stemming*
  end for
end for
Agglomerative Clustering
Dk←*distinct keywords*
for*i=0*: *Pd***do**
  for*j=0*: *Do***do**
    Dk← *select distinct keywords*
  end for
end for
Fid←*file ids*
for*i=0*: *Pd***do**
  for*j=0*: *Do***do**
    Fid ← *assign file id*
  end for
end for
Tf←*term frequency*
for*i=0*: *Pd***do**
  for*j=0*: *Do***do**
    Tf← *calculate term frequency*
  end for
end for
Ed←*euclidean distance*
while *true*
  for*i=0*: *Pd***do**
    for*j=0*: *Pd***do**

$$Ed = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots .. + (q_i - p_j)^2}$$

    end for
  end for
  Med←*minimum euclidean distance*
 Md← *merged data*
end while

DOI: 10.24818/18423264/52.2.18.11

Generate base point using ECDH
Cosine Similarity
Cs ← *cosine similarity*
for*i=0*: *Md***do**
  for*j=0*: *Dr***do**

$$Cs_{ij} = \frac{\sum_{k=1}^{n} A_k B_k}{\sqrt{\sum_{k=1}^{n} A_k^2}\sqrt{\sum_{k=1}^{n} B_k^2}}$$

  end for
end for
Rd ← *ranked data*

3. PERFORMANCE ANALYSIS

This section specifically exemplifies the performance of the proposed DAC-SSR in terms of Root Mean Square Error (RMSE), Mean Absolute Error (MAE), running time as an evaluation with the existing Online Graph Regularized user Preference Learning with Frank Wolfe algorithm (OGRPL-FW). In addition to this a comparative analysis between different sorts of recommendations systems is nailed with respect to manifold techniques such as Collaborative Filtering (CF), Content Based Recommendation (CBR), Hybrid Recommendation regarding running time with datasets acquired. Likewise, analysis of micro-F1 and macro F1measure is perceived in terms of training ratio for the aforementioned recommendation system formulating methodologies packet [20]. The benchmark dataset named as 20 news group dataset[31] with the domains regarding business, entertainment, politics, sports and technology is deliberated to authenticate the performance of devised DAC-SSR against the existing methods. The software requirements necessitated for realizing the proposed scenario with all its proficiency is through deploying it in a Windows 7 OS that possesses Java as a front end and MySql at its backend. The IDE being utilized is Eclipse Europa and the WAmp Server 2.0 serves as a Database. The compiler necessitated for executing the overall procedure is JDK 1.7.

3.1 Root Mean Square Error

The RMSE value is mathematically expressed as follows:

$$RMSE = \sqrt{\frac{1}{|\Omega_{test}|}\sum_{(i,j)\in\Omega_{test}}\left(r_{ij} - \widehat{r_{ij}}\right)^2} \quad (7)$$

Where, Ω_{test} – number of entries possessed by training data
r_{ij} – actual rating of j[th] user on i[th] item
$\widehat{r_{ij}}$ – Estimated rating of j[th] user on i[th] item

DOI: 10.24818/18423264/52.2.18.11

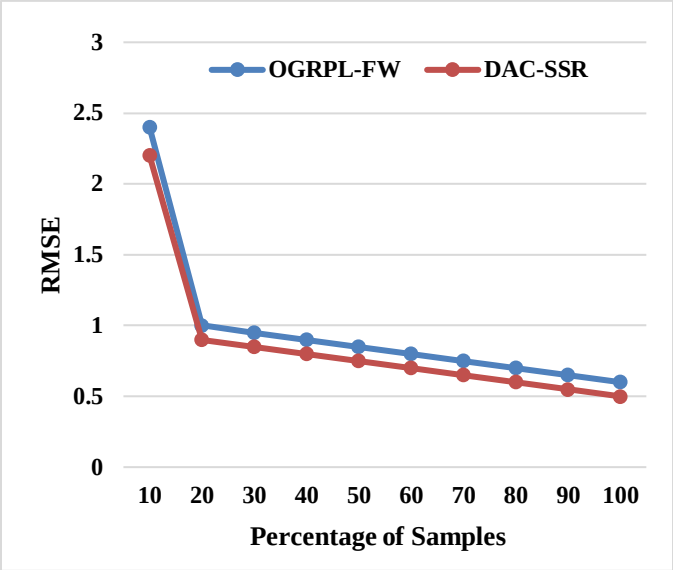

Figure 2. RMSE Analysis

Fig. 2 graphically plots the variation of RMSE with the percentage of samples from 10 to 100. The RMSE values for minimum and maximum samples are 2.4 and 0.6 respectively for OGRPL-FW method. The provision of distance-based clustering and the similarity measures in proposed DAC-SSR minimizes the RMSE values to 2.2 and 0.5 respectively. The comparative analysis between the proposed DAC-SSR and OGRPL-FW states that the proposed scheme offers 8.3 and 16.67 % for minimum and maximum samples respectively.

3.2 Mean Absolute Error

The mathematical formulation of MSE is expressed as follows:

$$MAE = \frac{1}{|\Omega_{test}|}\sum_{(i,j)\in\Omega_{test}}\left|r_{ij} - \widehat{r_{ij}}\right| \quad (8)$$

Where, Ω_{test} – number of entries possessed by training data

r_{ij} – actual rating of j[th] user on i[th] item

$\widehat{r_{ij}}$ – Estimated rating of j[th] user on i[th] item

Fig. 3 illustrates the MAE variations with respect to the minimum (10) and maximum (100) percentage samples respectively. The MAE values for minimum and maximum samples are 1.55 and 0.5 respectively for an OGRPL-FW method. The

DOI: 10.24818/18423264/52.2.18.11

provision of distance-based clustering and the similarity measures in proposed DAC-SSR minimizes the RMSE values to 1.3 and 0.5 respectively. The comparative analysis between the proposed DAC-SSR and OGRPL-FW states that the proposed scheme offers 16.13% for maximum samples respectively.

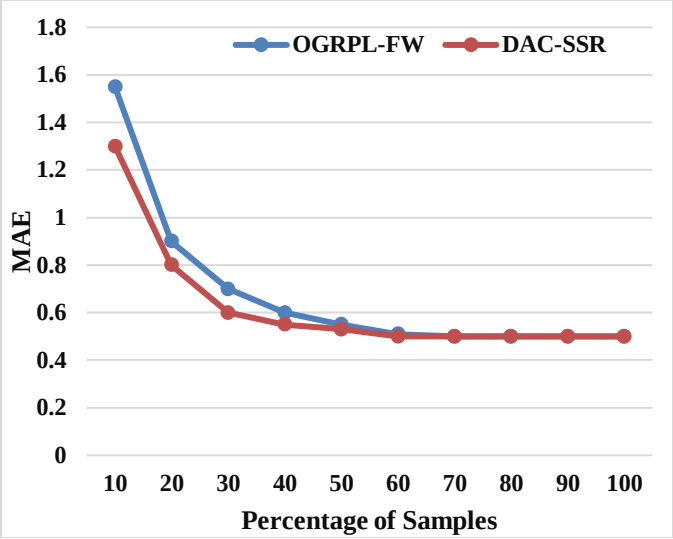

Figure 3. Mean Absolute Error

3.3 Running Time analysis

The time required to complete the overall process depends on the number of samples. In this section, the running time variations are investigated with respect to the percentage of samples and type of datasets used in Fig. 4 and Fig. 5 respectively.

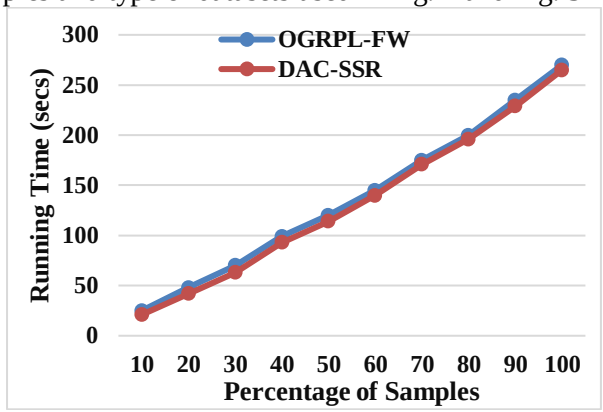

Figure 4. Running Time analysis between OGRPL-FW and DAC-SSR

DOI: 10.24818/18423264/52.2.18.11

The time consumption for minimum and maximum samples are25 and 270 secs respectively for OGRPL-FW method. The prior clustering and the similarity measures in proposed DAC-SSR minimizes the time consumption into 21 and 265 secs respectively. The comparative analysis between the proposed DAC-SSR and OGRPL-FW states that the proposed scheme offers 16 and 1.85 % for maximum samples respectively.

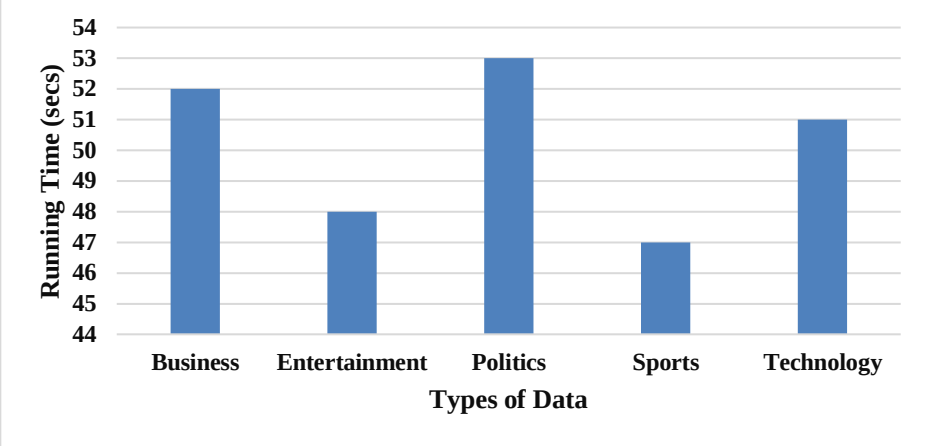

Figure 5. Running Time analysis of DAC-SSR for different datasets

The running time analysis with respect to different datasets shows that the time consumption for sports is minimum and politics is maximum due to the dimensionality.

The timespan utilized for completing those tasks stated via user defined queries with respect to ascending number of samples are compared in terms of two diverse scenarios incorporated with Hadoop framework and processing without Hadoop.Fig. 6 typically plots the time span accomplished with the devised DAC-SSR methodology.

DOI: 10.24818/18423264/52.2.18.11

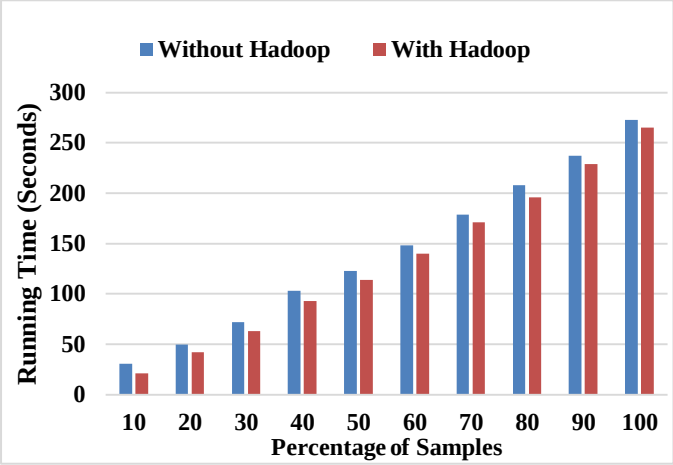

Figure 6. Running Time analysis of DAC-SSR incorporated with Hadoop framework and without Hadoop framework

The minimum samples account for 10 numbers while the maximum counts for 100. The time span inferred for minimum number of samples is 32% diminished when compared to the scenario deployed without Hadoop framework and the time span mitigation inferred for the maximum number of samples is depleted by 2%. The time consumed for processing the information without integrating Hadoop framework measures significantly higher than DAC-SSR assimilated with Hadoop framework owing to the optimizing functionality involved.

3.4 Micro-F1 analysis

The micro and macro-F1 analysis is illustrated in Fig. 7 and 8 show that the proposed approach offers better F1 values due to similarity-based clustering..

DOI: 10.24818/18423264/52.2.18.11

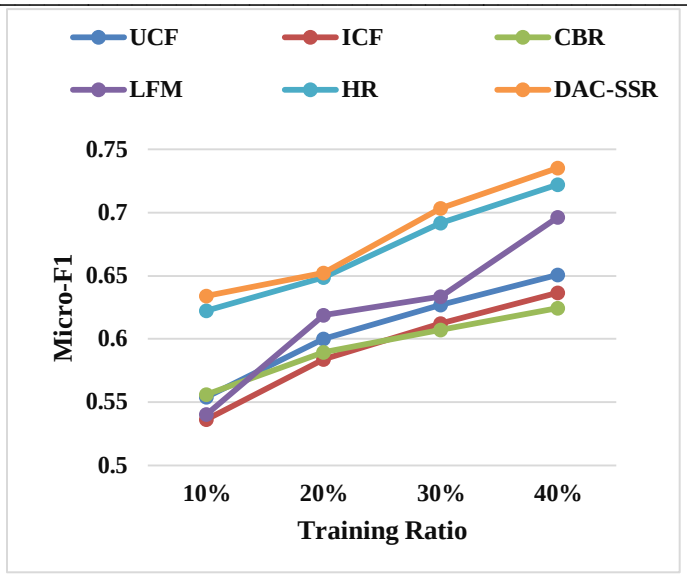

Figure 7. Micro-F1 analysis of DAC-SSR for different datasets

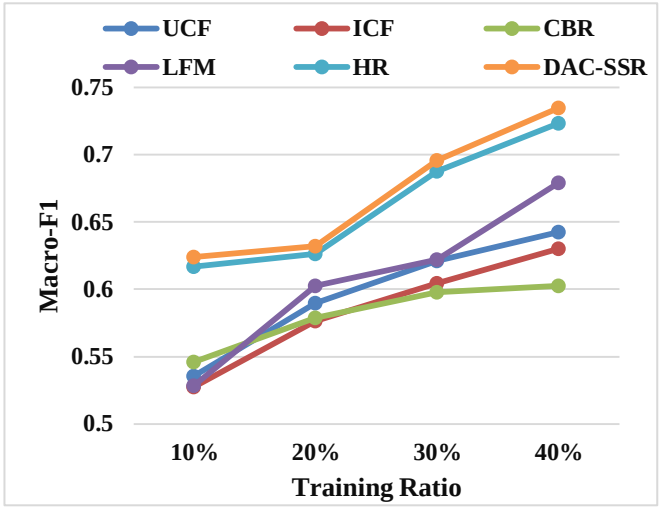

Figure 8. Macro-F1 analysis of the DAC-SSR for different datasets

DOI: 10.24818/18423264/52.2.18.11

The micro F1 analysis unveils the performance of utilizing a common labels by revealing the weights employed on all sorts of samples being utilized. The macro F1 is computed by performing a mean arithmetic calculation on the harmonic F1 measure acquired in prior. It accounts for the performance of the rare category of labels involved in processing.

4. CONCLUSION AND FUTURE WORK

In this paper, a novel secure web service recommendation system is devised through interpretation of similarity between subsets in the cloud environment. Primarily, the datasets that comprised of web services are processed through a progressive agglomerative clustering that adopts the Euclidean distance prevailing in between those preprocessed datasets. In addition, those tightly bonded clusters are assessed for their similarity on the basis of user-item through Tf-Idf assessment an item-item similarity relying upon computation of cosine similarity. A robust securing strategy termed as ECDH is employed in the development of DAC-SSR approach that keenly encrypts those stipulated web services ranked on the basis of their similarity being assessed. This entire setup is realized in Hadoop MapReduce framework and the proposed algorithm accomplished a performance enhancement in terms of running time by 1.85 % that sounds superior to other conventional CF methodologies for providing pertinent web services subjected to certain optimality constraints.

REFERENCES

[1]**J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang (2015),** ***Recommender System Application Developments: A Survey;*** *Decision Support Systems,* vol. 74, pp. 12-32;

[2]**Y. Hong, J. Vaidya, H. Lu, P. Karras, and S. Goel (2015), Collaborative** ***Search Log Sanitization: Toward Differential Privacy and Boosted Utility;*** *IEEE Transactions on Dependable and Secure Computing,* vol. 12, pp. 504-518;

[3]**Y. Xu and J. Yin (2015),*Collaborative Recommendation with User Generated Content;*** *Engineering Applications of Artificial Intelligence,* vol. 45, pp. 281-294;

[4]**Y. Cai, H.-f. Leung, Q. Li, H. Min, J. Tang, and J. Li (2014),** ***Typicality-based Collaborative Filtering Recommendation;*** *IEEE Transactions on Knowledge and Data Engineering,* vol. 26, pp. 766-779;

[5]**S.-Y. Lin, C.-H. Lai, C.-H. Wu, and C.-C. Lo (2014),** ***A Trustworthy QoS-Based Collaborative Filtering Approach for Web Service Discovery;*** *Journal of Systems and Software,* vol. 93, pp. 217-228;

DOI: 10.24818/18423264/52.2.18.11

[6]**E. Baccarelli, N. Cordeschi, A. Mei, M. Panella, M. Shojafar, and J. Stefa(2016),** ***Energy-Efficient Dynamic Traffic Offloading and Reconfiguration of Networked Data Centers for Big Data Stream Mobile Computing: Review, Challenges, and A Case Study*****;*** *IEEE Network,* vol. 30, pp. 54-61;

[7]**G. Bello-Orgaz, J. J. Jung, and D. Camacho (2016),** ***Social Big Data: Recent Achievements and New Challenges;*** *Information Fusion,* vol. 28, pp. 45-59;

[8]**V. Chang and M. Ramachandran (2016),*****Towards Achieving Data Security with the Cloud Computing Adoption Framework;*** *IEEE Transactions on Services Computing,* vol. 9, pp. 138-151;

[9]**S. Huang, J. Ma, P. Cheng, and S. Wang (2015),*****A Hybrid Multigroup Coclustering Recommendation Framework Based on Information Fusion****;* *ACM Transactions on Intelligent Systems and Technology (TIST),* vol. 6,p. 27;

[10]J. **Liu, Y. Jiang, Z. Li, X. Zhang, and H. Lu (2016),** ***Domain-Sensitive Recommendation with User-Item Subgroup Analysis*****;** *IEEE Transactions on Knowledge and Data Engineering,* vol. 28, pp. 939-950;

[11]**Z. Yang, B. Wu, K. Zheng, X. Wang, and L. Lei (2016),** ***A Survey of Collaborative Filtering-Based Recommender Systems for Mobile Internet Applications;*** *IEEE Access,* vol. 4, pp. 3273-3287;

[12]**M. Jiang, P. Cui, X. Chen, F. Wang, W. Zhu and S. Yang (2015),** ***Social Recommendation with Cross-Domain Transferable Knowledge****;IEEE Transactions on Knowledge and Data Engineering,* vol. 27, pp. 3084-3097;

[13]**A. Gani, A. Siddiqa, S. Shamshirband and F. Hanum (2016),** ***A Survey on Indexing Techniques for Big Data: Taxonomy and Performance Evaluation;*** *Knowledge and Information Systems,* vol. 46, pp. 241-284;

[14]**G. Kang, M. Tang, J. Liu, X. F. Liu, and B. Cao (2016),** ***Diversifying Web Service Recommendation Results via Exploring Service Usage History;*** *IEEE Transactions on Services Computing,* vol. 9, pp. 566-579;

[15]**X. Chen, Z. Zheng, Q. Yu, and M. R. Lyu (2014),** ***Web Service Recommendation via Exploiting Location and Qos Information****; IEEE Transactions on Parallel and Distributed Systems,* vol. 25, pp. 1913-1924;

[16] **S. Wang, Z. Zheng, Z. Wu, M. R. Lyu, and F. Yang (2015),** ***Reputation Measurement and Malicious Feedback Rating Prevention in Web Service Recommendation Systems****; IEEE Transactions on Services Computing,* vol. 8, pp. 755-767;

[17]**H. Hoffmann and M. Söllner (2014), I*****ncorporating Behavioral Trust Theory into System Development for Ubiquitous Applications****; Personal and ubiquitous computing,* vol. 18, pp. 117-128;

DOI: 10.24818/18423264/52.2.18.11

[18]**H. Sun, Z. Zheng, J. Chen and M. R. Lyu (2013),** ***Personalized Web Service Recommendation via Normal Recovery Collaborative Filtering*****;** *IEEE Transactions on Services Computing,* vol. 6, pp. 573-579;

[19]**Z. Wang, J. Liao, Q. Cao, H. Qi and Z. Wang (2015),** ***Friendbook: A Semantic-Based Friend Recommendation System for Social Networks****; IEEE Transactions on Mobile Computing,* vol. 14, pp. 538-551;

[20]**L. Yao, Q. Z. Sheng, A. H. Ngu, J. Yu and A. Segev (2015),** ***Unified Collaborative and Content-Based Web Service Recommendation*****;** *IEEE Transactions on Services Computing,* vol. 8, pp. 453-466;

[21]**J. Li, Y. Ji-Jiang, Y. Zhao, B. Liu, M. Zhou, J. Bi,** ***et al.*** **(2016),** ***Enforcing Differential Privacy for Shared Collaborative Filtering****; IEEE Access*;

[22]**J. Vera-del-Campo, J. Pegueroles, J. Hernández-Serrano and M. Soriano(2014),** ***Doccloud: A Document Recommender System on Cloud Computing with Plausible Deniability****; Information Sciences,* vol. 258, pp. 387-402;

[23]**T. R. Hoens, M. Blanton, A. Steele and N. V. Chawla (2013),** ***Reliable Medical Recommendation Systems with Patient Privacy;*** *ACM Transactions on Intelligent Systems and Technology (TIST),* vol. 4, p. 67;

[24]**E. Toch (2014),** ***Crowdsourcing Privacy Preferences in Context-Aware Applications*****;** *Personal and ubiquitous computing,* vol. 18, pp. 129-141;

[25]**P. Dandekar, N. Fawaz and S. Ioannidis (2014),** ***Privacy Auctions for Recommender Systems;*** *ACM Transactions on Economics and Computation,* vol. 2, p. 12;

[26]**J. Wu, L. Chen, Z. Zheng, M. R. Lyu and Z. Wu (2014),** ***Clustering Web Services to Facilitate Service Discovery;*** *Knowledge and information systems,* vol. 38, pp. 207-229;

[27]**G.-N. Hu, X.-Y. Dai, Y. Song, S.-J. Huang and J.-J. Chen (2016),** ***A Synthetic Approach for Recommendation: Combining Ratings, Social Relations and Reviews;*** *arXiv preprint arXiv:1601.02327*;

[28]**J. Bu, X. Shen, B. Xu, C. Chen, X. He and D. Cai (2016),** ***Improving Collaborative Recommendation via User-Item Subgroups****; IEEE Transactions on Knowledge and Data Engineering,* vol. 28, pp. 2363-2375;

[29]**Z. Zhao, H. Lu, D. Cai, X. He and Y. Zhuang (2016),** ***User Preference Learning for Online Social Recommendation****; IEEE Transactions on Knowledge and Data Engineering,* vol. 28, pp. 2522-2534;

[30]**M. Khan, Y. Jin, M. Li, Y. Xiang and C. Jiang (2016),** ***Hadoop Performance Modeling for Job Estimation and Resource Provisioning;*** *IEEE Transactions on Parallel and Distributed Systems,* vol. 27, pp. 441-454;

[31]"***20 Newsgroups data set," ed, 2010.***

DOI: 10.24818/18423264/52.2.18.11