International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

Ideal Keyword Match in a Big Data Application Using Keyword Aware Service Recommendation Method

Tumma Susmitha[1], D. Mythili[2]
[*1]Computer Science & Engineering
Vasavi College of Engineering
Hyderabad, India
[*] susmitha.vce@gmail.com
[2]Computer Science & Engineering
Vasavi College of Engineering
Hyderabad, India
Prabhupatil84@gmail.com

Abstract— The big data movement additionally influenced service recommender systems. The emergence of alternative providers has created a big research issue in providing clients with relevant suggestions for services they want. Service recommender systems have proven to be helpful tools that help users manage the multitude of services at their disposal and provide pertinent recommendations. Because the quantity of customers, services, and other online information is growing exponentially, service recommender systems function in a "Big Data" context. This poses serious challenges for these systems. In this work, we address these difficulties by contributing the following: This makes use of a collaborative filtering algorithm that is user-input driven. Keywords extracted from user reviews reflect their preferences here. Additionally, we apply it to Hadoop, a distributed computing framework that builds on Map Reduce for processing. by applying a collaborative filtering process that is user-based. In the proposed system, we are using a user-based Collaborative Filtering method. It also has similarities to the existing system. We consider both customer reviews and company rankings. We provide KASR, a method for keyword-aware service recommendation. Key words in KASR serve as indicators of users' preferences, and recommendations are produced by a user-based Collaborative Filtering algorithm. A domain thesaurus and keyword-candidate list are provided to help better understand the preferences of the customers. The active user indicates their choices by selecting keywords and preferences from the keyword-candidate list.

Keywords-*: KASR, Qos, CFA*

I. INTRODUCTION

Service recommender systems have been impacted by the big data trend as well. Providing customers with appropriate recommendations for services they prefer has become a significant research challenge due to the proliferation of alternative providers. It has been demonstrated that service recommender systems are useful tools that assist users in managing the abundance of services available to them and make relevant recommendations. Service recommender systems operate in a "Big Data" environment due to an exponential rise in the number of consumers, services, and other online information. This presents significant hurdles for these systems.

Objective

We tackle these issues in this study by making the following contributions: This uses an algorithm for collaborative filtering that is based on user input. Here, user preferences are indicated by keywords that were taken from their reviews. Furthermore, we put it into practice using Hadoop, a distributed computing platform that utilizes Map Reduce as its computational foundation. The Porter Stemmer algorithm is employed in the current system to power the review-based websites that this system uses. This system only took into account business ratings; user reviews were not taken into account. Porter Stemmer algorithm is one of the system's shortcomings. Another is that it filters the number of service-oriented websites based on user recommendation and business logic, but this method is inaccurate since it ignores user reviews. The majority of service recommender systems in use today, such restaurant guides and hotel reservation systems, have identical service ratings and user-facing service recommendation lists. They have not taken into account the various tastes of users and

have not complied with their specific needs. Here is an illustration of a hotel reservation.
By utilizing a user-based collaborative filtering algorithm, we are able to overcome ongoing projects in our system while prioritizing the viewpoint of the user over business logic.We are utilizing a user-based Collaborative Filtering algorithm in the suggested system. Additionally, it resembles the current system. We take into account both consumer feedback and business rankings. In collaborative filtering, keywords are used to indicate both of users' preferences and the quality of candidate services. Collaborative filtering on Map Reduce has favorable scalability and efficiency.

II. Related Work

In [1], numerous recommender systems use collaborative filtering (CF) methods often, however because of their high computational cost, their application in large-scale systems is constrained. In order to address the scalability issue of CF, we build the user-based CF algorithm on the Hadoop cloud computing platform in this study. According to the results of the experiments, a straightforward method that divides users into groups based on two fundamental ideas, namely, a neat arrangement of mapper numbers to avoid mapper initiation and partitioning tasks equally such that all processors finish tasks at the same time, can achieve linear speedup.

In [2], Authors suggest a recommendation system for online social networks in this paper that is based on Bayesian inference. Users can tell their friends how they rate the information in our system. Using a set of conditional probabilities obtained from their shared rating histories, a pair of friends' rating similarity is calculated. Through his direct and indirect acquaintances on the social network, a user spreads a content rating query. A Bayesian network is built to infer the rating of the inquiring user based on the query responses. We provide distributed protocols that online social networks can quickly implement. To address cold start and rating sparseness, we also suggest using prior distribution. Two distinct online user rating data sets are used to assess the suggested algorithm.

In [3], the abundance of computers in a way that makes cooperative problem-solving possible It has been suggested that general purpose computers with a generalized interconnection of memories or specialized computers with interconnections of memories that are geometrically connected and controlled by one or more instruction streams are the right directions. It is shown that the single processor strategy is still effective and that the multiple processor approach has limitations when applied to genuine issues and the anomalies that result from them.

In [4], Software programs called recommender systems make an effort to lessen information overload. Their objective is to suggest interesting products to end users depending on their preferences. Most recommender systems use collaborative filtering to do this. Parallel to this, the discipline of decision science known as multiple criteria decision analysis (MCDA) tries to model and analyze the decision maker's value system in order to aid in the decision-making process. The Collaborative Filtering strategy is combined with a hybrid framework that contains MCDA techniques, and these two approaches are evaluated in this article. Due to two key factors, the proposed methodology enhances the effectiveness of straightforward multi-rating recommender systems. First, groups of user profiles are created before the implementation of the user modeling procedure used to create these profiles, which is based on each user's value system and makes use of many criteria decision analysis techniques, and the collaborative filtering algorithm. The aforementioned assertion is supported by experiments using real user data.

In [5], an overview of the category of multi-criteria recommender systems is the goal of this chapter. The article examines MCD methods and approaches that can help the design of multi-criteria recommenders after first defining the recommendation problem as an MCDM (multi-criteria decision making) problem. It then concentrates on the group of techniques known as multi-criteria rating recommenders, which make suggestions by simulating a user's utility for a product as a vector of ratings along a number of criteria. Currently used multi-criteria algorithms for making predictions and producing recommendations are reviewed. The chapter comes to a close with a consideration of unresolved problems and potential difficulties for the class of recommenders for multi-criteria ratings.
In [6], one of the drivers for the Semantic Web's creation has always been semantic search. We offer a framework for utilizing ontology-based knowledge bases to enhance search across big document collections. In our interpretation of information retrieval on the semantic web, a search engine responds to user requests by returning documents instead of, or in addition to, specific values. Our strategy involves a retrieval system and an ontology-based plan for the semi-automatic annotation of texts. An annotation weighting algorithm and a ranking algorithm are components of the retrieval model, which is based on an adaptation of the traditional vector-space model. Tolerating knowledge base

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

incompleteness is accomplished by combining semantic search with traditional keyword-based retrieval.

In [7], an all-purpose distributed execution engine called Dryad is used for applications that use coarse-grained data parallelism. A dataflow graph is created by a Dryad application by fusing computational "vertices" with communication "channels". In order to launch the program, Dryad executes the nodes of this graph on a selection of accessible computers, corresponding as necessary via shared-memory FIFOs, TCP pipelines, and files. The vertices offered by the application developer are often written as sequential programs that don't use thread creation or locking and are relatively simple. Dryad's scheduling of vertices to execute concurrently on various machines or on various CPU cores inside a same computer results in concurrency. In order to make the most effective use of the available resources, the application can determine the size and location of data at runtime and adjust the graph as computation moves forward. Dryad is built to grow from robust single PCs with several cores, to modest computer clusters, to data centers with thousands of computers. The Dryad execution engine takes care of all the challenging issues involved in building a sizable distributed, concurrent application, including scheduling CPU usage, recovering from network or computer failures, and moving data across vertices.

In [8], For processing and producing big data sets, Map Reduce is a programming concept and its related implementation. Users define a reduction function, which merges all intermediate values associated with the same intermediate key, and a map function, which splits a key/value pair into a collection of intermediate key/value pairs. As demonstrated in the research, this model can convey a variety of activities from the actual world. This functional programming technique automatically parallelizes and runs programs on a big cluster of common machines. The run-time system manages the requirements for inter-machine communication, handles machine failures, schedules the execution of the program across a number of machines, and partitions the input data as needed.

III. METHODOLOGY

In this work, we've created a working prototype that allows live VM migration between any two nodes on the Internet, even if they're connected by separate networks. Future research will focus on fault-tolerance support for a SOC system. Sensitivity analysis will be done to determine how violating our model assumptions may affect the best way to allocate resources.

User Interface Design

Users must provide their username and password in order to connect to the server, and only then will they be permitted to do so. If a user already has an account, they can log in directly; otherwise, they must register their username, password, and email address with the server. In order to maintain the upload and download rates, the server will create an account for each user. User ID will be set to Name. Typically, logging in is required to access a particular website.

Quality Of Service

Each front-end server that proxy servers receives data requests from customers for this module. After receiving the data, a sense is made automatically to determine whether there are any other servers on the server. And it relies on dynamically generated DNS replies, HTTP redirects, or tunneling requests through permanent HTTP proxies. We presume that each request source has a proxy and DNS server nearby.

Query analyzer

on this model, we take the user's query and analyze it to determine what the user needs. Then, using a map-reduce technique, we search through numerous cloud databases at once to find the pertinent data on a number of virtual machines.

KASR

Key word aware service recommendation is referred to as KASR. In this model, we compute the value of the user and produce the result after obtaining the value from the qos model and satisfying the user's demand.

CFA

Collaborative filtering algorithm is referred to as CFA. We take the value from the previous model and use the algorithm to filter the value in the running time environment in the cloud environment according to priority in this algorithm. The CF algorithm is based on user input and is utilized to produce suitable suggestions. KASR seeks to provide a user with a personalized service recommendation list that includes recommendations for the best services based on a personalized rating of each candidate service.

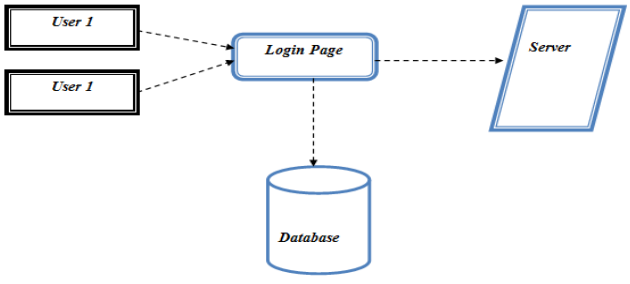

Fig. 1. User Interface Design

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9
Article Received: *25 July 2023* ***Revised:*** *12 September 2023* ***Accepted:*** *30 September 2023*

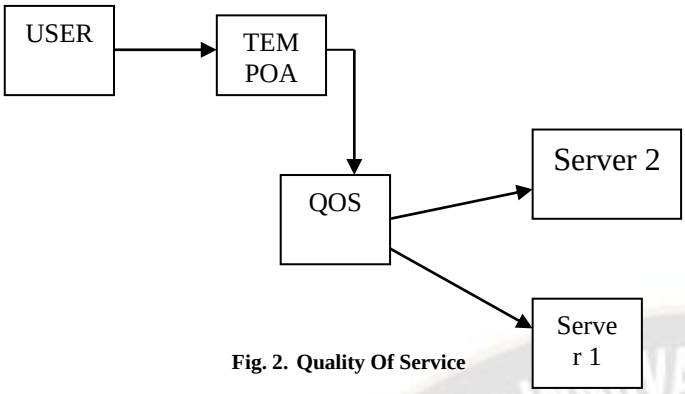

Fig. 2. Quality Of Service

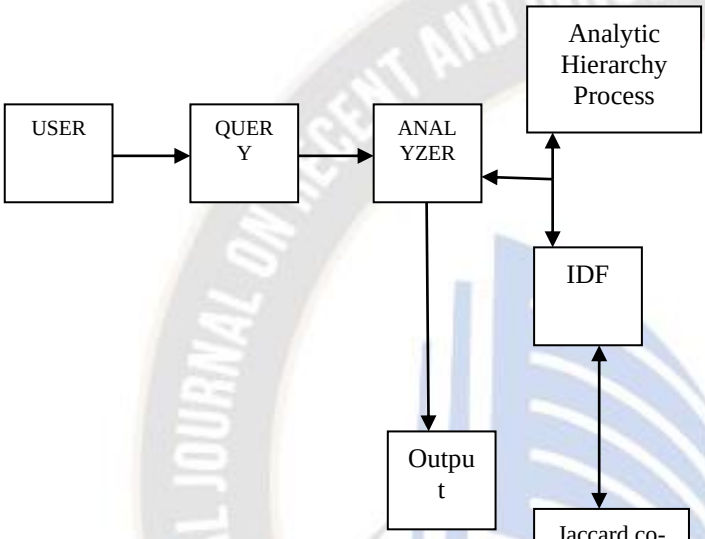

Fig. 3. Query analyzer

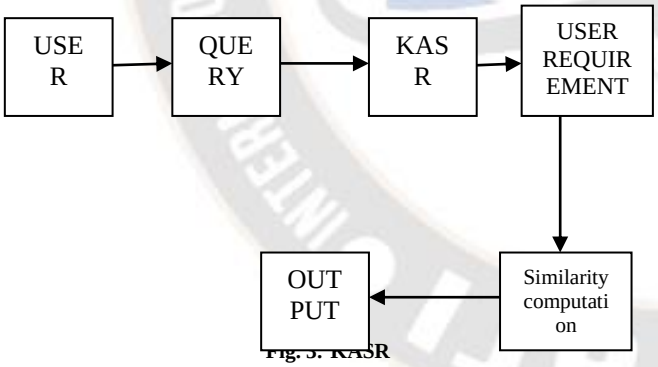

Fig. 3. KASR

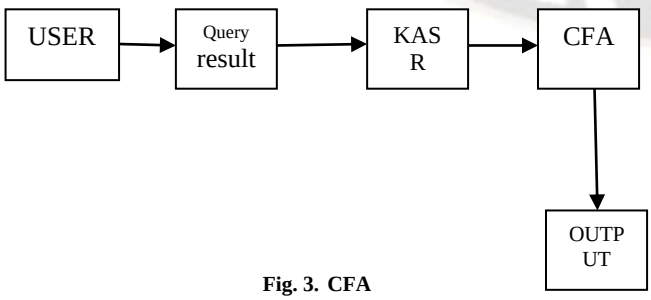

Fig. 3. CFA

Keyword aware service recommendation method

We have put forth KASR, a keyword-aware service recommendation technique. In KASR, users' preferences are indicated via key phrases, and recommendations are generated through a user-based Collaborative Filtering algorithm. To further understand customers' preferences, a keyword-candidate list and domain thesaurus are offered. By choosing keywords and preferences from the keyword-candidate list, the active user expresses his or her preferences. The systems architect creates the fundamental foundation for the system, outlining the fundamental core design elements and features that serve as the cornerstone for everything else and are the most difficult to modify afterwards. The systems architect works to preserve the integrity of the users' vision as it changes through thorough design and execution, giving the architects a view of what the system has to be and accomplish, as well as the paths along which it must be able to progress.

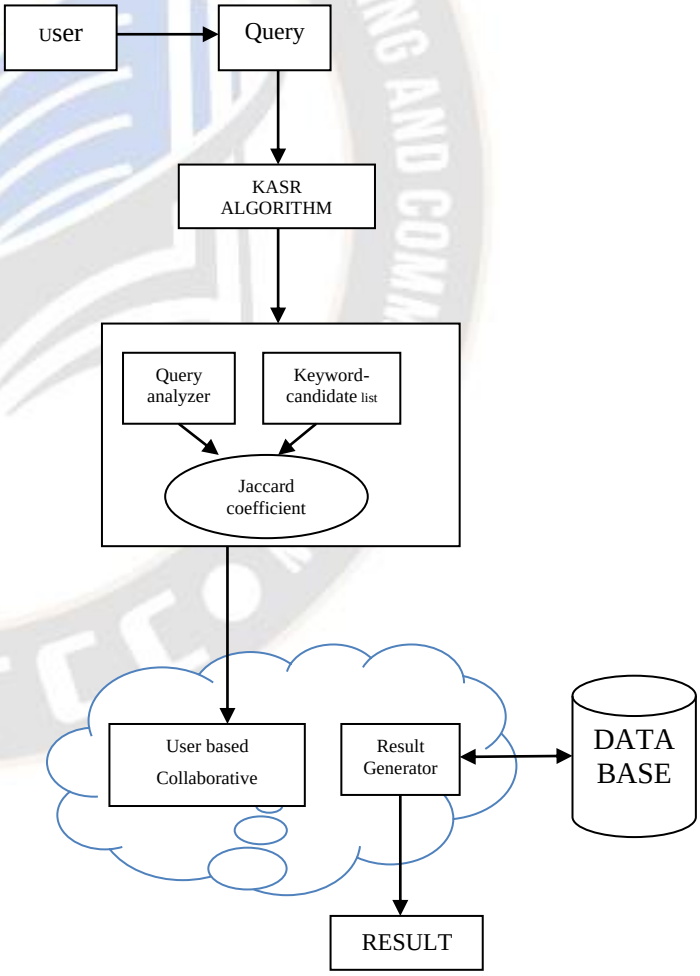

Fig. 3. Architectural Diagram

IV. Conclusion And Future Work

The solution we have suggested in this research is called KASR, which is a keyword-aware service recommendation system. In KASR, users' preferences are expressed using key words, and relevant recommendations are produced using a user-based Collaborative Filtering algorithm. More precisely, to assist in gathering users' preferences, a keyword-candidate list and domain thesaurus are supplied. The current user indicates his or her preferences by choosing keywords from the list of potential keywords, and the preferences of past users can be gleaned from their evaluations of the services in accordance with the domain thesaurus and keyword-candidate list. Our approach seeks to provide users with a tailored service suggestion list and suggest the service or services that would be most suitable for them.

References

[1] J. Manyika, M. Chui, B. Brown, et al, "Big Data: The next frontier for innovation, competition, and productivity," 2011.

[2] C. Lynch, "Big Data: How do your data grow?" Nature, Vol. 455, No. 7209, pp. 28-29, 2008.

[3] F. Chang, J. Dean, S. Ghemawat, and W. C. Hsieh, "Bigtable: A distributed storage system for structured data," ACM Transactions on Computer Systems,Vol. 26, No. 2 (4) , 2008.

[4] W. Dou, X. Zhang, J. Liu, J. Chen, "HireSome-II: Towards Privacy-Aware Cross-Cloud Service Composition for Big Data Applications," IEEE Transactions on Parallel and Distributed Systems, 2013.

[5] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet, Vol. 7, No.1, pp. 76-80, 2003.

[6] M. Bjelica, "Towards TV Recommender System Experiments with User Modeling," IEEE Transactions on Consumer Electronics, Vol. 56, No.3, pp. 1763-1769, 2010.

[7] M. Alduan, F. Alvarez, J. Menendez, and O. Baez, "Recommender System for Sport Videos Based on User Audiovisual Consumption," IEEE Transactions on Multimedia, Vol. 14, No.6, pp. 1546-1557, 2013.

[8] Y. Chen, A. Cheng and W. Hsu, "Travel Recommendation by Mining People Attributes and Travel Group Types From Community-Contributed Photos". IEEE Transactions on Multimedia, Vol. 25, No.6, pp. 1283-1295, 2012.

[9] Z. Zheng, X Wu, Y Zhang, M Lyu, and J Wang, "QoS Ranking Prediction for Cloud Services," IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 6, pp. 1213-1222, 2013.

[10] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, "Recommending and Evaluating Choices in a Virtual Community of Use," In CHI '95 Proceedings of the SIGCHI Conference on Human Factors in Computing System, pp. 194-201, 1995.