Original Article
ISSN (Online): 2582-7472

ShodhKosh: Journal of Visual and Performing Arts
June 2024 5(6), 1691–1699

A REVIEW OF ENHANCE CODE QUALITY AND DEVELOPMENT EFFICIENCY BY BIG DATA INFRASTRUCTURE

Aarati Chavan[1], Dr. Uruj Jaleel[2], Dr. Nisha Auti[3]

[1]Research Scholar, Department of Computer Science and Engineering, Kalinga University, Raipur
[2]Department of Computer Science and Engineering, Kalinga University, Raipur
[3]Department of Computer Science and Engineering, Bharati Vidyapith, Pune

Corresponding Author
Aarati Chavan,
aratisawant85@gmail.com

DOI
10.29121/shodhkosh.v5.i6.2024.2495

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ABSTRACT

In the rapidly evolving landscape of software development, the need for high code quality and efficient development processes is paramount. The integration of Big Data infrastructure into software development workflows has emerged as a powerful approach to enhancing both code quality and development efficiency. This review explores how Big Data technologies can be leveraged to analyze vast amounts of code, identify patterns, predict potential bugs, and optimize development practices. This review underscores the transformative potential of Big Data in revolutionizing software development. The use of recommendation systems while developing software increasing in order to speed up the process of software development by software developers. Accurate recommendations leads to successful, faster, efficient development, but inaccurate recommendations can lead to inappropriate, missed deadline software development.

Keywords: Recommendations, Software, Big Data, Techniques, Mobile

1. INTRODUCTION

Computer programming is a method for solving problems by creating projects that can be executed. The designers of a piece of software take into account the needs of the application and put them into action using a programming language like Java, C#, Visual Basic.NET (VB.NET), etc. Learning computer programming is challenging, and many students struggle to do well in this area. How engaged students are with the material and their ability to retain it are both impacted by a variety of factors. The reasons why students have trouble grasping programming concepts have been investigated by experts. Some attribute the problems to the necessary massive knowledge of several fields, such as the ability to think critically and have a deep understanding of different programming languages. Others have hypothesised that the high percentage of programming course dropouts is due to software engineers' inadequate understanding of how programs work, which in turn makes it more difficult for students to grasp programming concepts.

In large software firms, the issue of mistake resolution becomes quite pressing since programmers devote a lot of time and effort studying documents to change. For example, as shown in Eclipse bug report #261613, the software engineer spent three days exhaustively searching through random data before making changes to just two pages. The programmers in bug report #241244 spent two weeks discussing the "main drivers" and their examination, with statements like "... I'd get a kick out of the chance to investigate this course further" and "Facilitate examination still required..." These examples demonstrated that programmers may significantly cut down on time spent on programming advancement projects if they could find documents to modify more effectively.

In the field of software engineering, this presents a significant obstacle. Very little has been proposed as of late to address these issues. For instance, the primary assemble has mined programming modification tales, and analysts have developed history-based proposal frameworks to aid programmers by following two ideal models. Prescription records are subject to change based on data mining programming updates. By exploring association relationships between records that have been edited jointly as often as possible, these approaches create document to-alter ideas.

APPROACHES OF RECOMMENDER SYSTEM

Below we described the all approaches of recommender system.

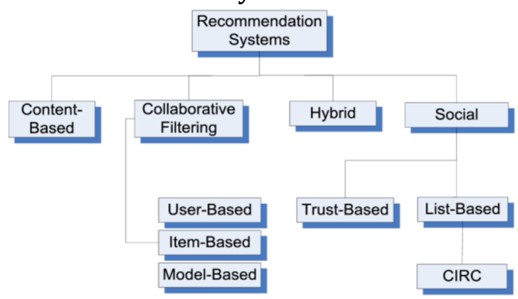

Figure 1: Approaches of Recommender System

Mobile recommender systems

Versatile recommender frameworks are one emerging area of research in the recommender frameworks field. It is now possible to provide personalised, setting-sensitive recommendations because to the increasing availability of online access to smart mobile phones. Because portable data is more complex than the data that recommender frameworks typically deal with (it is heterogeneous, explosive, requires spatial & transient auto-connection, and has approval and all-inclusive statement issues), this area of research presents unique challenges. One issue with adaptive recommender systems is that recommendations may not have much of an effect in some areas; for instance, it would be silly to advise a dish in a place where most of the components would not be accessible.

An example of a portable recommender system would be one that provides city taxi drivers with potentially useful driving lessons. To determine the routes that cabbies took while on the clock, this system takes data inputs including GPS coordinates (latitude & longitude), time stamps, & operational status (with or without passengers). By analysing this data, it may generate a list of pickup spots along a route, which improves occupancy times and benefits. Minimal computational and energy needs are essential for this framework because of its apparent area subordination & need to run on a portable or implanted device.

Things made for expert clientele are another example of the mobile recommendation in action. Based on the client's situation and interests, it suggests relevant info using GPS indications and his plan. In order to continuously fine-tune the adaptable recommender framework for the benefit of the client, the framework employs machine learning methodologies & thinking patterns. The algorithm's developer dubbed it hybrid-ε- greedy.

The "Internet of Information" has also proved useful in the successful construction of mobile recommendation systems by providing a hub for organised data. As an example of a respectable framework, SMARTMUSEUM Even when provided with very little client data, the system is able to provide content-coordinating client premiums by using semantic display, data recovery, & machine learning techniques.

RISK-AWARE RECOMMENDER SYSTEMS

Most existing methods for recommender systems disregard the possibility of client discomfort in certain scenarios in favour of prioritising content delivery based on logical facts. If you want to avoid annoying your customer, it's best to avoid pushing suggestions at certain times and places, including during an expert conference, first thing in the morning, or late at night. This applies to numerous applications, like providing customised content. How well the recommender system has integrated risk into the recommendation process is, thus, a limited determinant of its performance.

RISK DEFINITION

"The risk in suggestion systems is the potential to disturb or to disturb the user resulting to a bad answer of the user".

A dynamic risk-mindful recommender system (DRARS) was developed by the authors of DRARS, A Dynamic Risk-Mindful Recommender Framework, in response to these issues. DRARS represents the setting-mindful recommendation as a bandit problem. A logical bandit algorithm & substance-based technique are combined in this framework. They proved that DRARS improves upon the best algorithm currently available, the Upper Certainty Bound (UCB), by determining the optimal investigation value to maintain a trade-off between investigation & abuse given the current client's risk level. The researchers demonstrated through real-world tests using actual data & users that the proposed frameworks performed far better when users' risk levels were included.

PERFORMANCE MEASURES

To determine whether recommendation algorithms are adequate, evaluation is crucial. Mean squared error & root mean squared error are the most used metrics; the latter was used in the Netflix Prize. When evaluating a recommendation method, it is helpful to use data recovery metrics like accuracy and review or DCG. Good diversity, interest, and scope are now also seen as crucial evaluation aspects. Still, several of the model evaluation tools are severely criticised. The results of purportedly independent evaluations don't always line up with the actual satisfaction of questioned customers. Finally, "we would propose treating consequences of disconnected assessments with scepticism" according to the authors.

MULTI-CRITERIA RECOMMENDER SYSTEMS

Multi-criteria recommender frameworks (MCRS) can be characterized as recommender frameworks that join inclination data upon numerous criteria. Rather than creating suggestion strategies in light of solitary measure values, the general inclination of client u for the thing I, these frameworks endeavour to anticipate a rating for unexplored things of u by misusing inclination data on different criteria that influence this general inclination value. A few analysts approach MCRS as a multi-criteria basic leadership (MCDM) issue and apply MCDM strategies and methods to actualize MCRS frameworks. Get this section for an expanded presentation.

OPINION-BASED RECOMMENDER SYSTEMS

Customers are rarely given the option to provide comments or surveys on the products they purchase. Because they contain the client's evaluation of the object as well as their thoughts & feelings about it, the client-created writings provide the recommender system with credible information. Because they also reflect elements of the object, like meta-information, deleted highlights are widely worried by the customers, and improved meta-information of things is what the highlights extracted from client-produced surveys are. The client's evaluation ratings on the related highlights may be seen as the estimates subtracted from the surveys. Content mining, data recovery, & opinion evaluation are some of the well-known methods used in presumption-based recommender frameworks.

RECOMMENDER SYSTEM EVALUATION

The success of the measure is evaluated using implicit metrics like click-through rate or conversion rate. Offline assessments rely on previously collected data, such as a dataset detailing consumers' movie rating history. Next, we evaluate recommendation methods by seeing how well they forecast users' ratings in the dataset. Although ratings reveal how much a user enjoyed a film, this data isn't always accessible. Take citation recommender systems as an example; users usually don't give ratings to citations or suggested articles. Offline assessments in these situations could make use of implicit effectiveness metrics.

Take the case of a recommender framework that can prescribe an unlimited number of articles for an exploratory article's reference list as an example. It may be considered successful. However, many experts consider this type of independent evaluations to be fundamental. Evidence suggests, for instance, that client considerations or A/B tests contribute to the poor correlation between disconnected evaluation results and actual outcomes. A dataset that is commonly used for disconnection evaluation has been found to include duplicate data, leading to incorrect algorithm evaluation results.

REPRODUCIBILITY IN RECOMMENDER SYSTEM RESEARCH

In recent times, there has been a growing realisation within the network regarding the lack of impact of previous research on practical application of recommender systems. According to Ekstrand, Konstan, et al., "it is as of now hard to duplicate & expand recommender frameworks enquire about outcomes" and evaluations are "not taken care of reliably." "The Recommender Frameworks enquire about network is confronting an emergency where numerous present results that contribute little to aggregate learning [...] regularly in light of the fact that the investigation does not have the [...] assessment to be effectively judged and, subsequently, to give important contributions," Konstan and

Adomavicius speculate. Consequently, a large body of literature about recommender frameworks is likely not replicable. As a result, recommender system administrators find little guidance in the flow inquiry on taking note of the inquiry, which suggests approaches to handle users in recommender systems. Said and Bellogín oversaw an examination of current articles published in the area, compared the most popular proposal systems, and found significant inconsistencies in revenue distribution, even when using the same algorithms and data sets. A small number of experts demonstrated that substantial changes in the recommender framework's feasibility occurred in response to small modifications in the proposal algorithms or scenarios. Seven actions are critical to improving the momentum situation because: "(1) description other research fields and gain from them, (2) locate a typical comprehension of reproducibility, (3) recognise and recognise the determinants that influence reproducibility, (4) direct more thorough investigations (5) modernise production rehearses, (6) cultivate the improvement and utilisation of proposal structures, and (7) build up best-hone rules for recommender-frameworks research."

Engineers that create software spend a lot of time looking for files that they can change. To illustrate the developer's methodology, consider Overshadowing bug report#261613. It shows that for three days, the developer extensively examined irrelevant documents before making edits to just two entries. The programmer voiced their: "I believe I'm getting closer to the genuine reason for the situation...." Comparatively, in bug report #241244, developers had a discourse over an examination of "underlying drivers" for two weeks, writing:" Further examination still required…," and "… I'd get a kick out of the chance to research this course further." Time spent on programming progress errands would be significantly reduced, according to these examples, if developers could more readily find documents to edit.

Analysts have used two paradigms to build history-based recommendation frameworks that programmers may use. Mining programming revision histories is the main aggregate's forte. One example is the suggestion to update prescription records based on data gathered from programming. In order to provide record-to-alter suggestions, these methods mine association rules between documents based on how frequently they may have been edited together in the past. The history of developer interactions has been mined by the second group. Based on the software engineer's communication history that was mined, it was suggested, among other things, to prescribe tactics or records to view.

BIG DATA

Big data refers to data sets that are so large and complicated that traditional methods of data handling application programming are unable to deal with them. Major informational obstacles include data collection, storage, analysis, querying, search, interchange, representation, questioning, refreshment, data security, and data source. Initially, there were three concepts associated with massive amounts of data: volume, assortment, & speed. Various theories later associated with massive amounts of data are fact. These days, the term "huge data" is more often used to describe large data sets used in predictive analytics, customer behaviour analysis, or other advanced data mining techniques that extract value from large datasets, rather than a specific data size. "The volumes of data that are currently available are undeniably massive, yet that is hardly the most important standard in this new information ecosystem. New linkages can be discovered through the examination of data sets in order to "spot business patterns, anticipate infections, or battle wrongdoing etc." Web look, finance, urban informatics, and corporate informatics are only a few of the areas where researchers, government authorities, and industry professionals sometimes face difficulties when dealing with massive informative indexes. Ecology, meteorology, genomics, sophisticated materials science reproductions, and association omics are just a few of the e-Science fields where researchers encounter limitations.

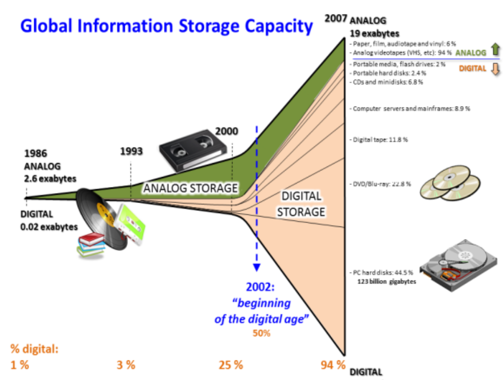

Figure 2 History of Global Information Storage Capacity

Informational collections develop quickly in part because they are progressively accumulated by modest and various data detecting Web of things gadgets, for example, cell phones, airborne (remote detecting), programming logs, cameras, mouthpieces, radio-recurrence ID (RFID) perusers and remote sensor systems. The world's mechanical per-capita ability to store data has generally multiplied at regular intervals since the 1980s; starting at 2012, consistently 2.5 exabytes (2.5×1018) of information is created.

Depend on an IDC report forecast; the worldwide information volume will develop exponentially from 4.4 zetta bytes to 44 zetta bytes in the vicinity of 2013 and 2020. By 2025, IDC predicts there will be 163 zetta bytes of information. One inquiry for vast endeavours is figuring out who should possess huge information activities that influence the whole association.

When dealing with massive amounts of data, relational database management systems, business intelligence tools, and image data programming packages often run into problems. It could call for "big parallel programming running on tens, hundreds, or even a large number of servers". "Big data" definitions change depending on client and device capabilities, and expanding capabilities make massive data an ever-evolving target. Some organisations may need to reevaluate their data management decisions when suddenly faced with terabytes of data. Although other people could need tens or even hundreds of terabytes of data before they find data measuring useful.

Although the word has been in usage since the 1990s, its proponents point to John Mashey as the one who popularised it. Data sets that are too large for commonly used programming tools to capture, manage, and process in a reasonable amount of time are typically considered enormous data sets. Although all three kinds of data fall under massive information theory, the real action takes place with unstructured data. Many Exabytes of data to a few hundred terabytes, the "estimate" of massive amounts of data is an ever-changing focal point, beginning in 2012. To decipher pieces of knowledge from diverse, complicated, and massively scaled datasets, a plethora of innovative tactics & innovations must be implemented.

According to a 2016 definition, "Big data speaks to the data resources defined as with sufficient volume, speed and assortment requiring particular innovation and systematic strategies for its change into value" . In addition, some groups append the letter V for truth in order to depict revisionism, which has been tested by several professionals in the field. Additional comparable properties of big data that have been expanded upon include the three Vs: volume, variety, and speed.

- **Machine learning:** large data frequently doesn't probe for reasons but rather identifies patterns.
- Big data is frequently an unpaid byproduct of digital engagement; it leaves a digital footprint.

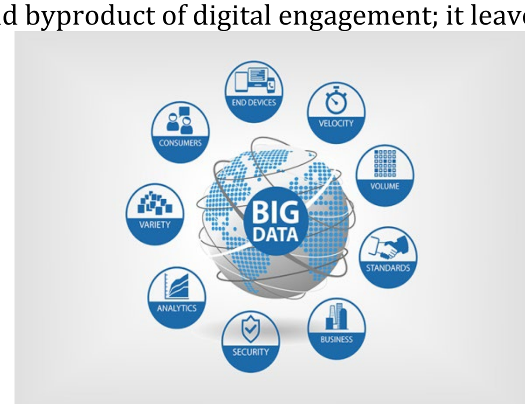

Figure 3: Components of Big Data

A 2018 definition states "Big data is the place parallel processing devices are expected to deal with information", and notes, "This speaks to an unmistakable and obviously characterized change in the software engineering utilized, through parallel programming hypotheses, and misfortunes of a portion of the certifications and abilities made by Codd's relational model."

The growing concept further illustrates the variance between "big data" & "Business Intelligence":

- Business Knowledge utilizes unmistakable insights with information with high data thickness to quantify things, identify patterns, and so forth.
- Big information utilizes inductive measurements and ideas from nonlinear framework recognizable proof to derive laws (relapses, nonlinear connections, and causal impacts) from expansive arrangements of information with low data thickness to uncover connections and conditions or to perform expectations of results and practices.

BIG DATA CHARACTERISTICS

There are a number of ways to characterise big data:

- **VOLUME:** The amount of data that is created & saved. Data size is a defining factor in data value, insight potential, & big data classification.
- **VARIETY:** The data's genre & characteristics. That way, those doing the analysis will be able to put the knowledge they gain to good use. Data fusion allows big data to fill in gaps, and it pulls from many sources, including text, photos, audio, & video.
- **VELOCITY:** The velocity of data generation and processing in this context is crucial for meeting the needs and overcoming the obstacles that development & growth entail. A lot of the time, big data is accessible in real.
- **VERACITY:** The precision of the analysis is affected by the wide variation in the data quality of the collected data.

There could be a 6C system in cyber-physical systems & factory labour:

- Connection (sensor & networks).
- Cloud (computing & data on demand).
- Cyber (model & memory).
- Content/context (meaning & correlation).
- Community (sharing & collaboration).
- Customization (personalization & value).

Identifying meaningful data requires the use of state-of-the-art equipment (analysis & algorithms). A good example of this is the need to think about both visible and invisible problems with various components when handling a manufacturing line. On the factory floor, data age algorithms need to detect and fix invisible problems like machine corruption, segment wear, and so on.

TECHNOLOGIES

- Analytics methods include A/B testing, ML, & NLP; Big data tools including BI, cloud, and databases; and Data analysis techniques
- Graphical representations of the data, including charts & graphs

Big data with several dimensions, sometimes known as tensors, can be better handled using tensor-based algorithms like multi linear subspace learning. Additional developments associated with big data include web-based applications, distributed record frameworks, appropriated databases, applications based on massively parallel processing (MPP) databases, information mining, the cloud, and HPC-based foundations (applications, storage, and registration assets). Despite many new approaches and developments, doing machine learning with massive data is still challenging.

Petabytes of data may be stored & managed by some MPP social databases. Using the massive RDBMS data tables to their full potential requires the ability to stack, screen, descend, & enhance.

With the 2008 launch of a company named Ayasdi, the technology developed by DARPA's Topological Information Examination program which seeks the fundamental structure of massive informational indexes was made public.

As a whole, professionals in big data analysis are worried about slower shared capacity and are leaning towards direct-attached storage (DAS) in all its forms, from solid-state drives (SSDs) to high-capacity SATA plates housed in parallel handling hubs. The common perception of storage area networks (SANs) and network attached storage (NASs) is that they are moderately complex, expensive, & time-consuming. Huge data investigation frameworks that flourish with framework execution, product infrastructure, & minimal cost do not exhibit these features consistently.

One of the defining features of big data analysis is real or near continuous data transmission. This has led people to avoid being inactive whenever possible. Knowledge stored in memory is excellent, but knowledge stored on the other end of an FC SAN association isn't. When compared to alternative capacity processes, the cost of a SAN at the scale needed for investigational applications is significantly greater.

2. LITERATURE REVIEW

Ali Davoudian et al. (2020) An up-and-coming category of scalable software systems, Big Data Systems (BDSs) enable the management, analysis (in batch, stream, or hybrid mode), and presentation of large volumes of diverse data to both internal users & external applications. Such systems may grow exceedingly complicated due to changes in data, technology, & intended use, and thus provide unique difficulties across the whole software development lifecycle. Because of this, the adoption of BDSs has been challenging for many organisations and businesses. We shed light on three main software engineering tasks within the framework of BDSs & decisions taken to address them in light of current research and industry initiatives. Included in this category are tasks like software/data quality assurance, engineering requirements, and developing & building software to fulfil these needs. In addition, we bring to light a few outstanding difficulties in creating efficient BDSs that require the joint efforts of academics & industry professionals.

Han Hu et al. (2014) In the last twenty years, there has been an explosion of data from many sources due to technology improvements. This data comes from a wide variety of sources, including sensors used in healthcare & science, user-generated data, the Internet, financial institutions, & supply chain management systems. To describe this new development, the word "big data" was created. When contrasted with more conventional forms of data, big data stands out not just for its massive size but also for other distinctive features. Big data, for example, often lacks organisation & necessitates examination in real-time. A major paradigm shift in data collection, transfer, storage, & processing at scale is required to accommodate this new information. With the hope of giving laypeople a bird's-eye view & inspiring more technical readers to get their hands dirty building their own big data analytics platforms, we offer a literature review & system tutorial in this article. We begin with a definition of large data and a discussion of the difficulties associated with it. Data production, data collection, data storage, & data analytics make up the four consecutive modules that make up big data systems. Here is a methodical approach to break them down. An important data value chain consists of these four components. After that, we give a comprehensive overview of several methods & techniques from the academic and business worlds. Further, we introduce the widely used Hadoop framework for handling large data issues. Our paper concludes by outlining several assessment standards and future areas of research for large data systems.

Radwa Elshawi et al. (2018) Our capacity to handle, analyse, & comprehend the massive amounts of data produced and collected by nearly all of our activities, as well as the technological advances that have allowed us to do so, have been undergoing remarkable growth in recent years. What we now call "Big Data Science" is where these tendencies meet. Data storage and processing systems that can scale are essential for Big Data Science. To facilitate the storage, processing, & advanced analytics applications that rely on Big Data, cloud computing offers a realistic and economical alternative. We take a close look at the components of the software stack that enable data scientists to use Big Data Science as a commodity service. Furthermore, we examine and categorise the cutting-edge big data analytics frameworks, which are mostly hosted on Clouds, according to the service models that they offer. In addition, we offer a variety of perspectives on the most recent continuing discoveries and unanswered questions in this field.

Ahmed Oussous et al. (2018) The importance of developing Big Data applications has grown in recent years. Actually, data mining has become more important for a variety of businesses across industries. Nevertheless, conventional data methods & systems perform poorly when applied to Big Data. Their reactivity is poor, and they aren't scalable, fast, or accurate. A lot of effort has been put into tackling the complicated Big Data problems. Different distributions & technologies have emerged as a consequence. The latest innovations in Big Data technology are summarised in this review article. The goal is to assist in choosing & implementing the appropriate mix of Big Data technologies based on one's technical requirements and the demands of one's particular applications. In addition to a high-level overview of the most important Big Data technologies, it also compares them according to several system levels, including storage, processing, querying, access, & management. Features, benefits, limitations, and applications of key technologies are categorised and discussed.

Nawsher Khan et al. (2014) Many people in the academic & IT sectors are interested in Big Data. Information is created and gathered in the computer and digital realms at a rate that quickly surpasses the limit range. More than 2 billion people throughout the globe have access to the web, and more than 5 billion people have cell phones. Forecasts indicate that 50 billion devices will have Internet connectivity by 2020. Present projections indicate data output will be 44 times higher than 2009 levels. Both the amount of data and the rate of market growth are increasing as a result of the increased speed of information transport and sharing over optic fibre and wireless networks. Security, data diversity, transmission speed, & ever-increasing volume of data are just a few of the issues brought about by the exponential expansion of big data. No comprehensive assessment of the field has been conducted, and Big Data is still in its early stages. Therefore, this research examines and categorises all the many aspects of Big Data, such as its definitions, nature, volume, management, analysis, & security, as well as its fast development rate and quick growth rate. A data life cycle based on Big Data concepts and terminology is also suggested in this research. The prospects and several unanswered questions in Big Data dominance dictate the future of this field's study. These lines of inquiry pave the way for more domain investigation & creation of more effective methods to deal with Big Data.

Azlinah Mohamed et al. (2020) Due to the creation of vast data created from many sources such multimedia apps, the IoT, & social media, big data has become an important study subject. Many decision making & forecasting sectors have been greatly impacted by big data, including recommendation systems, business analysis, healthcare, online display advertising, physicians, transportation, fraud detection, & tourist marketing. The distribution, communication, and processing of the massive amounts of data has been made possible by the quick development of several big data technologies in the academic and industry communities, including Hadoop, Storm, Spark, Flink, Kafka, or Pig. In order to analyse massive volumes of data efficiently, big data applications employ big data analytics techniques. However, owing to the difficulties in processing & utilising big data, it is difficult to choose the appropriate big data tools based on analytics approaches for developing a big data system. Researchers & practitioners working on big data systems lack sufficient knowledge of the platform's present technologies and requirements. Discussions about the benefits and drawbacks of big data technology, as well as practical approaches to addressing Big Data issues, are so necessary. Consequently, this study offers a literature review that examines the applications of big data analytics methods & technologies in several domains, including healthcare, social media, public sector, economics, and business. This paper aims to(1) gain an understanding of the current state of big data research and (2) determine patterns in the utilise or study of big data tools based on batch or stream processing and big data analytics methods. (3) provide guidance and support to new researchers or practitioners in appropriately placing their research activity in this domain. The study's results will shed light on the current state of big data platforms & fields in which they are most useful, as well as on the pros and cons of big data tools, analytics methods, and applications, and on potential avenues for future research into the advancement of big data systems.

3. CONCLUSION

The ability of recommender systems to provide users with useful resource suggestions is a significant benefit. Discovering new and surprising recommendations may be a breeze with it. The recommender system's ability to locate engaging materials is dependent on the algorithm it employs. the integration of Big Data infrastructure into software development processes offers a significant opportunity to enhance both code quality and development efficiency. The use of real-time analytics and machine learning not only accelerates the development cycle but also ensures higher reliability and maintainability of software products. As the field continues to evolve, ongoing research and innovation will be essential to fully realize the potential of Big Data in transforming software development practices.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

Ricardo Tubio ; Rafael Sotelo ; Yolanda Blanco ; Martin Lopez ; Alberto Gil ; Jose Pazos ; Manuel Ramos, "A TV-anytime metadata approach to TV program recommendation for groups", 2008 IEEE International Symposium on Consumer Electronics, 2008.

Janez Zaletelj; Richard Wages; Tobias Burger; Stefan M. Grunvogel, "Content Recommendation System in the Production of Multi-Channel TV Programs", Third International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution (AXMEDIS'07), 2007.

Yu Yan; Kohei Hara; Takenobu Kazuma; Aiguo He, "A Method for Personalized C Programming Learning Contents Recommendation to Enhance Traditional Instruction", 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA), 2017.

Carlos Becerra ; Roberto Muñoz ; René Noël ; Marta Barría, "Learning objects recommendation platform based on learning styles for programming fundamentals",2016 XI Latin American Conference on Learning Objects and Technology (LACLO), 2016.

Fatmah Yousef Assiri, "Recommendations to improve programming skills of students of computer science", 2016 SAI Computing Conference (SAI), 2016.

Mengyi Zhang ; Minyong Shi ; Zhiguo Hong ; Songtao Shang ; Menghan Yan, “A TV program recommendation system based on big data”, 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), 2016.

Davoudian, Ali, and Mengchi Liu. "Big data systems: A software engineering perspective." ACM Computing Surveys (CSUR) 53.5 (2020): 1-39.

Hu, Han, et al. "Toward scalable systems for big data analytics: A technology tutorial." IEEE access 2 (2014): 652-687.

Elshawi, Radwa, et al. "Big data systems meet machine learning challenges: towards big data science as a service." Big data research 14 (2018): 1-11.

Mavrogiorgos, Konstantinos, et al. "Self-Adaptable Infrastructure Management for Analyzing the Efficiency of Big Data Stores." Journal of Advances in Information Technology Vol 13.5 (2022).

Oussous, Ahmed, et al. "Big Data technologies: A survey." Journal of King Saud University-Computer and Information Sciences 30.4 (2018): 431-448.

Khan, Nawsher, et al. "Big data: survey, technologies, opportunities, and challenges." The scientific world journal 2014.1 (2014): 712826.

Jiangshan Xu ; Liang-Jie Zhang ; Haiming Lu ; Yanda Li, “The development and prospect of personalized TV program recommendation systems”, Fourth International Symposium on Multimedia Software Engineering, 2002. Proceedings, 2002.

Jinshui Wang ; Xin Peng ; Zhenchang Xing ; Kun Fu ; Wenyun Zhao, “Contextual Recommendation of Relevant Program Elements in an Interactive Feature Location Process”,2017 IEEE 17th International Working Conference on Source Code Analysis and Manipulation (SCAM), 2017.

Fulian Yin ; Xiaowei Liu ; Wanying Ding ; Ruizhe Zhang, “Tag-Based collaborative filtering recommendation algorithm for TV program”, 2016 13th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), 2016.