REVOLUTIONIZING HEALTH SYSTEMS: MACHINE LEARNING-DRIVEN SOFTWARE ENGINEERING

[1] B.Srinivasulu, Assistant Professor, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad.
[2] Y.Jasmine, B.Tech, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad.
[3] V.Victoria Rachel, B.Tech, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad.
[4] V.Ravika, B.Tech, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad.
[5] Y.Sowmya, B.Tech, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad.

Abstract: Recently, machine learning has become a hot research topic. Therefore, this study investigates the interaction between software engineering and machine learning within the context of health systems. We proposed a novel framework for health informatics: the framework and methodology of software engineering for machine learning in health informatics (SEMLHI). The SEMLHI framework includes four modules (software, machine learning, machine learning algorithms, and health informatics data) that organize the tasks in the framework using a SEMLHI methodology, thereby enabling researchers and developers to analyze health informatics software from an engineering perspective and providing developers with a new road map for designing health applications with system functions and software implementations. Our novel approach sheds light on its features and allows users to study and analyze the user requirements and determine both the function of objects related to the system and the machine learning algorithms that must be applied to the dataset. Our dataset used in this research consists of real data and was originally collected from a hospital run by the Palestine government covering the last three years. The SEMLHI methodology includes seven phases: designing, implementing, maintaining and defining workflows; structuring information; ensuring security and privacy; performance testing and evaluation; and releasing the software applications.

1. INTRODUCTION

The field of health informatics (HI) aims to provide a large scale linkage among disparate ideas. Normally, a healthcare dataset is found to be incomplete and noisy; as a result, reading data from dataset linkage traditionally fails within the discipline of software engineering. Machine learning (ML) is a rapidly maturing branch of computer science since it can store data on a large scale. Many ML tools can be used to analyze data and yield knowledge that can improve the quality of work for both staff and doctors; however, for developers, there is currently no methodology that can be used. Regarding software engineering, there has been a lack of approaches to evaluating which software engineering tasks are better performed by automation and which require human involvement or human-in-the-loop approaches [1]. Big data has many challenges regarding analysis challenges for real-world big data [2], including OLAP mass data, mass data protection, mass data survey and mass data dissemination. Recently, a set of frameworks have been used to develop data analysis tools such as Win-CASE [3] and SAM [4]. The market has vast data analysis tools that can discover interesting patterns and hidden relationships to support deci sion makers [5]. BKMR used the R package as a statisti cal approach on health effects to estimate the multivariable exposure-response function [6]. Augmentor included the Python image library for aug mentation [7], while for the visualization of medical treat ment plans and patient data, CareVis was used [8], as it was designed for this task. Other applications require a visual interface using COQUITO [9]. For health-care data analytics, the widely known 3P tools [10] were used. Many simple applications, such as WEKA, which provided a GUI for many machine learning algorithms [11], while Apache Spark was used for the cluster computing framework [12], are powerful systems that can used in various applications for solving prob lems using big data and machine learning [13]. Table 1 sum marizes the main tools used for big data in analytics according with respect to the task. Software engineering for machine learning applications (SEMLA) discusses the challenges, new insights, and practical ideas regarding the engineering of ML and artificial engineering (AI) [14]. NSGA-II proposed algorithms for real-world applications that include more than one objective function for enhancing performance in terms of both diversity and convergence [15]. ML algorithms in clinical genomics generally come in three main forms: super vised, unsupervised and semi-supervised [16]. Interflow sys tem requirement analysis (ISRA) has been used to determine the system requirements [17]. Electronic healthcare (eHealth) frameworks have replaced traditional medical frameworks to improve mobile health care (mHealth) and enable patient-to-physician and patient to-patient interactions to achieve improved healthcare and quality of life (QoL) [18]. Big data and IoT have been used for improving the efficiency of m-health systems by predicting potential life-threatening conditions during the early stages [19]. Intelligent IoT eHealth solutions enable healthcare

professionals to monitor health-related data con tinuously and provide real-time actionable insights used to support decision making [20]. Machine learning is a field of software engineering that frequently utilizes factual procedures to enable PCs to ‘‘learn’’ by using information from saved datasets. Unsu pervised or information mining focuses more on exploratory information investigation and is known as learning supported by data analytics. Patient laboratory test queue management and wait time prediction are a challenging and complicated job. Because each patient might require different phase oper ations (tasks), such as a check-up, various tests, e.g., a sugar level test or blood test, X-rays or surgery, each task can consider different medical tests, from 0 to N, for each patient according to their condition. In this article, based on a grounded theory method ology [21], the researchers proposed a novel methodol ogy, SEMLHI, in developing a framework by defining the research problem and methodology for the developers. The SEMLHI framework includes a theoretical framework to support research and design activities that incorporate exist ing knowledge. The SEMLHI framework was composed of four components that help developers observe the health application flow from the main module to submodules to run and validate specific tasks. This enables multiple developers to work on different modules of the application simultane ously. The SEMLHI framework supports the methodological approach to conducting research on health informatics. It also supports a structure that presents a common set of ML terminology to use, compare, measure, and design software systems in the area of health. This creates a space whereby SE and ML experts can work on a specific methodological approach to enable health informatics software development teams to integrate the ML model lifecycle. Our methodology was applicable to current systems or in the development of new systems that use the ML module for current systems, which can be used in regular updates to add data to the system, to perform irregular updates and to add new features such as new versions of ICD diagnosis codes, minor model improvements for bug fixes, new functionalities required by the client, and new hardware or architectural constraints.

2. LITERATURE SURVEY

2.1 Interactive machine learning: Experimental evidence for the human in the algorithmic loop

ABSTRACT: Recent advances in automatic machine learning (aML) allow solving problems without any human intervention. However, sometimes a human-in-the-loop can be beneficial in solving computationally hard problems. In this paper we provide new experimental insights on how we can improve computational intelligence by complementing it with human intelligence in an interactive machine learning approach (iML). For this purpose, we used the Ant Colony Optimization (ACO) framework, because this fosters multi-agent approaches with human agents in the loop. We propose unification between the human intelligence and interaction skills and the computational power of an artificial system. The ACO framework is used on a case study solving the Traveling Salesman Problem, because of its many practical implications, e.g. in the medical domain. We used ACO due to the fact that it is one of the best algorithms used in many applied intelligence problems. For the evaluation we used gamification, i.e. we implemented a snake-like game called Traveling Snakesman with the MAX–MIN Ant System (MMAS) in the background. We extended the MMAS–Algorithm in a way, that the human can directly interact and influence the ants. This is done by “traveling” with the snake across the graph. Each time the human travels over an ant, the current pheromone value of the edge is multiplied by 5. This manipulation has an impact on the ant’s behavior (the probability that this edge is taken by the ant increases). The results show that the humans performing one tour through the graphs have a significant impact on the shortest path found by the MMAS. Consequently, our experiment demonstrates that in our case human intelligence can positively influence machine intelligence. To the best of our knowledge this is the first study of this kind.

2.2 Big data challenges and achievements: Applications on smart cities and energy sector

ABSTRACT: In this paper, the Big Data challenges and the processing is analyzed, recently great attention has been paid to the challenges for great data, largely due to the wide spread of applications and systems used in real life, such as presentation, modeling, processing and large (often unlimited) data storage. Mass Data Survey, OLAP Mass Data, Mass Data Dissemination and Mass Data Protection. Consequently, we focus on further research trends and, as a default, we will explore a future research challenge research project in this area of research.

2.3 CASE: A framework for computer supported outbreak detection

ABSTRACT: In computer supported outbreak detection, a statistical method is applied to a collection of cases to detect any excess cases for a particular disease. Whether a detected aberration is a true outbreak is decided by a human expert. We present a technical framework designed and implemented at the Swedish Institute for Infectious Disease Control for computer supported outbreak detection, where a database of case reports for a large number of infectious diseases can be processed using one or more statistical methods selected by the user. Based on case information, such as diagnosis and date, different statistical algorithms for detecting outbreaks can be applied, both on the disease level and the subtype level. The parameter settings for the algorithms can be configured independently for different diagnoses using the provided graphical interface. Input generators and output parsers are also provided for all supported algorithms. If an outbreak signal is detected, an email notification is sent to the persons listed as receivers for that particular disease. The framework is available as open source software, licensed under GNU General Public License Version 3. By making the code open source, we wish to encourage others to contribute to the future development of computer supported outbreak detection systems, and in particular to the development of the CASE framework.

2.4 Supporting iterative cohort construction with visual temporal queries

ABSTRACT: Many researchers across diverse disciplines aim to analyze the behavior of cohorts whose behaviors are recorded in large event databases. However, extracting cohorts from databases is a difficult yet important step, often overlooked in many analytical solutions. This is especially true when researchers wish to restrict their cohorts to exhibit a particular temporal pattern of interest. In order to fill this gap, we designed COQUITO, a visual interface that assists users defining cohorts with temporal constraints. COQUITO was designed to be comprehensible to domain experts with no preknowledge of database queries and also to encourage exploration. We then demonstrate the utility of COQUITO via two case studies, involving medical and social media researchers.

2.5 Healthcare analysis in smart big data analytics: Reviews, challenges and recommendations

ABSTRACT: Increasing demand and costs for healthcare is a challenge because of the high populations and the difficulty to cover all patients by the available doctors. The healthcare data processing and management became a challenge because the problems with the data itself like irregularity high-dimensionality, and sparsity. A number of researchers worked on these problems and provided some efficient and scalable healthcare solutions. we present the algorithms and systems for healthcare analytics and applications and some related solutions. The solution what we propose is depending on adding a new layer as middleware between the sources of heterogeneous data and the Map reduce Hadoop cluster. The solution solved the common problems of dealing with heterogeneous data effectively.

3. EXISTING SYSTEM

Normally, a healthcare dataset is found to be incomplete and noisy; as a result, reading data from dataset linkage traditionally fails within the discipline of software engineering. Machine learning (ML) is a rapidly maturing branch of computer science since it can store data on a large scale. Many ML tools can be used to analyze data and yield knowledge that can improve the quality of work for both staff and doctors; however, for developers, there is currently no methodology that can be used. Regarding software engineering, there has been a lack of approaches to evaluating which software engineering tasks are better performed by automation and which require human involvement or human-in-the-loop approaches.

DISADVANTAGES OF EXISTING SYSTEM

- Regarding software engineering, there has been a lack of approaches to evaluating which software engineering tasks are better performed by automation and which require human involvement or human-in-the-loop approaches..

4. PROPOSED SYSTEM

In propose paper author is combining Software Engineering and Machine Learning algorithms to improve disease prediction in health care systems and to minimize time taken to predict disease as we don't have enough hospitals or bed to accommodate growing number of patients and we can solve this problem of predicting disease with less time by employing software and machine learning algorithms. Propose paper concept is known as SEMLHI (where SE refers to software and ML refers to machine learning and HI refers to health data).

ADVANTAGES

- Advance machine learning algorithm called EXTREME LEARNING MACHINE (EML) and this EML algorithm is giving better prediction result compare to propose paper algorithms.

SYSTEM ARCHITECTURE

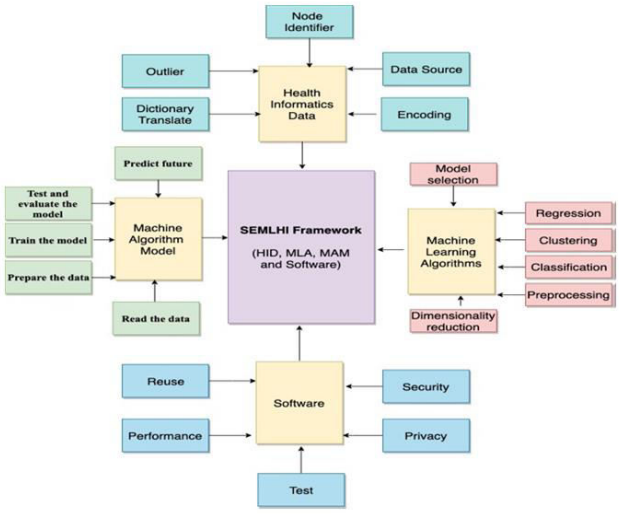

Fig 1: System Architecture

5. DATASET:

We can collect the dataset from the kaggle.com site and placed into our project folder.

	Number_pregnant	Glucose_concentration	Blood_pressure	Triceps	Insulin	BMI	Pedigree	Age	Class	Group
2	6	0.743718593	0.590163934	0.353535354	0	0.500745156	0.23441503	50	1	B
3	1	0.427135678	0.540983607	0.292929293	0	0.396423249	0.116567037	31	0	C
4	8	0.91959799	0.524590164	0	0	0.347242921	0.253629377	32	1	B
5	1	0.447236181	0.540983607	0.232323232	0.111111111	0.418777943	0.038001708	21	0	B
6	0	0.688442211	0.327868852	0.353535354	0.19858156	0.642324888	0.943637916	33	1	C
7	5	0.582914573	0.606557377	0	0	0.381520119	0.052519214	30	0	A
8	3	0.391959799	0.409836066	0.323232323	0.104018913	0.461997019	0.072587532	26	1	C
9	10	0.57788947	0	0	0	0.526080477	0.023911187	29	0	A
10	2	0.989949749	0.573770492	0.454545455	0.641843972	0.454545455	0.034158839	53	1	D
11	8	0.628140704	0.786885246	0	0	0	0.065755764	54	1	A
12	4	0.552763819	0.754098361	0	0	0.560357675	0.04824936	30	0	D
13	10	0.844221106	0.606557377	0	0	0.566318927	0.195986336	34	1	C
14	10	0.698492462	0.655737705	0	0	0.403874814	0.581981213	57	0	C
15	1	0.949748744	0.491803279	0.232323232	1	0.448584203	0.136635354	59	1	A
16	5	0.834170854	0.590163934	0.191919192	0.206855792	0.384500745	0.217335611	51	1	B
17	7	0.502512563	0	0	0	0.44709389	0.173356106	32	1	A
18	0	0.592964824	0.68852459	0.474747475	0.271867612	0.682563338	0.201964133	31	1	B
19	7	0.537688442	0.606557377	0	0	0.441132638	0.075149445	31	1	B
20	1	0.51758794	0.245901639	0.383838384	0.098108747	0.645305514	0.044833476	33	0	A
21	1	0.577889447	0.573770492	0.303030303	0.113475177	0.515648286	0.192570453	32	1	A
22	3	0.633165829	0.721311475	0.414141414	0.277777778	0.585692996	0.267292912	27	0	C
23	8	0.497487437	0.68852459	0	0	0.52757079	0.1323655	50	0	C
24	7	0.984924623	0.737704918	0	0	0.59314456	0.159265585	41	1	B
25	9	0.59798995	0.655737705	0.353535354	0	0.43219076	0.078992314	29	1	C
26	11	0.718592965	0.770491803	0.333333333	0.172576832	0.545454545	0.075149445	51	1	D
27	10	0.628140704	0.573770492	0.262626263	0.135933806	0.463487332	0.054227156	41	1	D
28	7	0.738693467	0.62295082	0	0	0.587183308	0.076430401	43	1	D
29	1	0.487437186	0.540983607	0.151515152	0.165484634	0.345752608	0.174637062	22	0	A
30	13	0.728643216	0.672131148	0.191919192	0.130023641	0.330849478	0.071306576	57	0	C
31	5	0.587939698	0.754098361	0	0	0.508196721	0.11058924	38	0	A

Table 5.1: Health data

6. UML DIAGRAMS

1. CLASS DIAGRAM

The cornerstone of event-driven data exploration is the class outline. Both broad practical verification of the application's precision and fine-grained demonstration of the model translation into software code rely on its availability. Class graphs are another data visualisation option.

The core components, application involvement, and class changes are all represented by comparable classes in the class diagram. Classes with three-participant boxes are referred to be "incorporated into the framework," and each class has three different locations:

• The techniques or actions that the class may use or reject are depicted at the bottom.

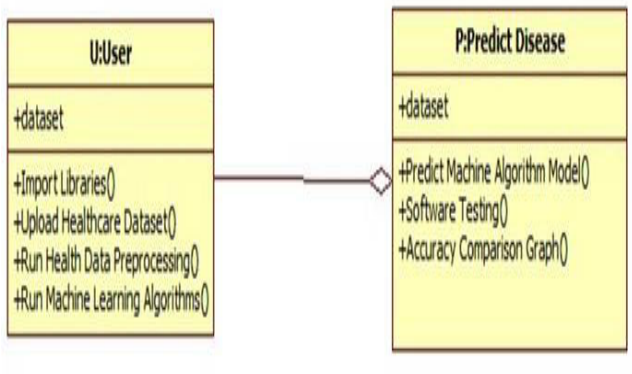

Fig 6.1 shows the class diagram of the project

2. USECASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

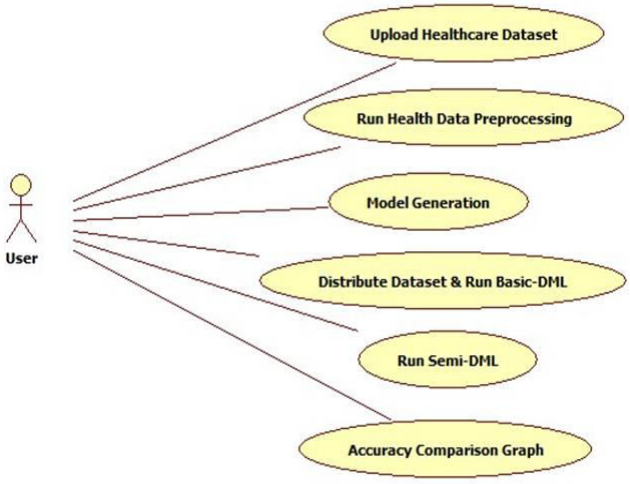

Fig 6.2 Shows the Use case Diagram

3. SEQUENCE DIAGRAM:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

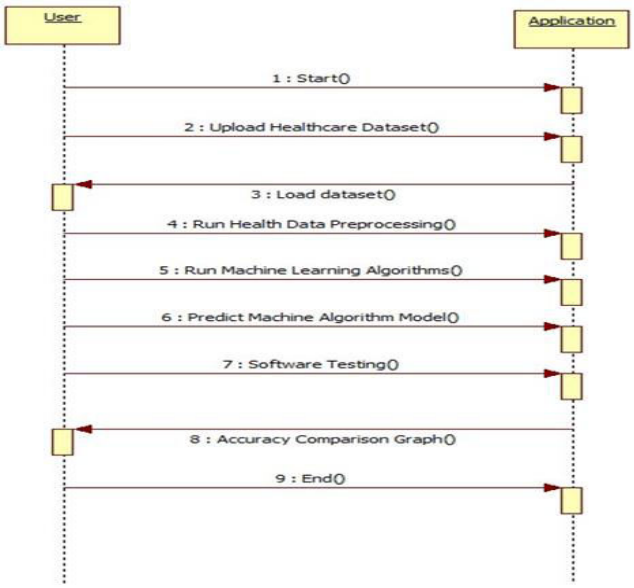

Fig 6.3 Shows the Sequence Diagram

7. RESULTS

7.1 Output Screens

Fig 7.1 Upload the Dataset

In above screen click on 'Upload Train Dataset' button and upload dataset

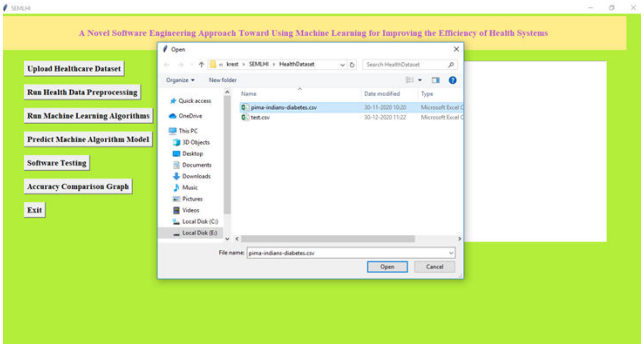

Fig 7.2 Uploading the Dataset File

In above screen selecting and uploading diabetes dataset and then click on 'Open' button to load dataset and to get below

Fig 7.3 Preprocess the dataset

In above graph in top we can see names of columns and in boxes values with minus symbols are not important and only positive column values are important and ML algorithm will train only with positive values and now close above graph to get below screen

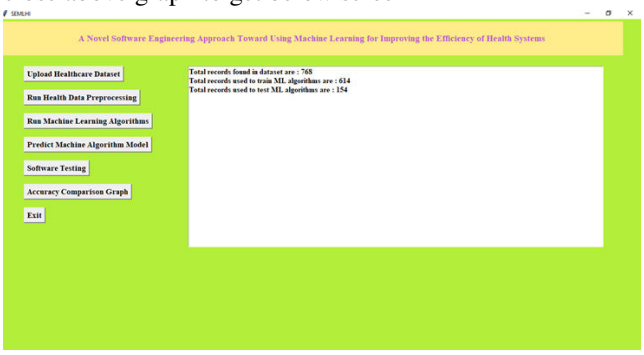

Fig 7.4 Data Preprocessing

In above screen after applying pre-processing and PCA we got total records as 768 and application using 614 records to train ML algorithms and to generate model and then used 154 records to test that trained model and to calculate prediction accuracy. Now both train and test data is ready and now click on 'Run Machine Learning Algorithms' button to start training all ML algorithms on train and test data

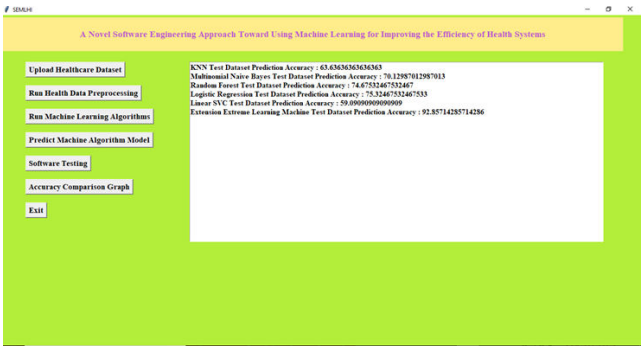

Fig 7.5 Run the Machine Learning Algorithms

In above screen we can see prediction accuracy of each algorithm and from all algorithms extension Extreme

Machine Learning is giving good prediction accuracy and now all ML algorithms are ready with trained model and now click on 'Predict Machine Algorithm Model' button to upload new test records and then ML will predict whether new test records contains positive or negative disease

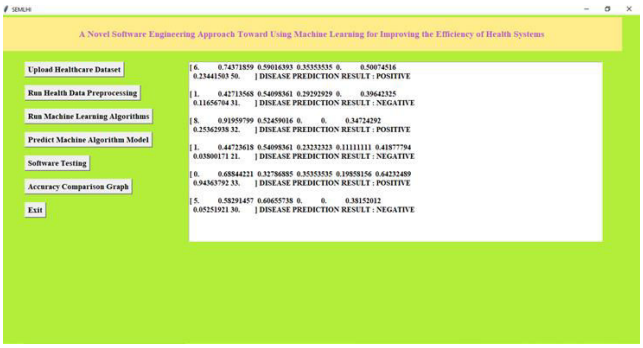

Fig 7.6 Accuracy of ML Algorithms

In above screen for each test lab record ML predict whether disease is positive or negative. Now click on 'Accuracy Comparison Graph' button to get below graph

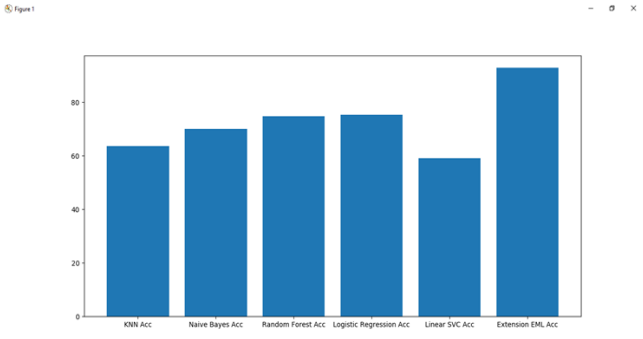

Fig 7.7 Accuracy Comparison Graph

In above graph x-axis represents ML algorithm names and y-axis represents accuracy of all those algorithms and from above graph we can conclude that extension EML is giving better accuracy

8. CONCLUSION

This article addressed an important HI with ML topic in software engineering by proposing an efficient new method approach related to software engineering, identified in prior research studies, using original data sets collected during the last 3 years from a Palestine hospital. This methodology allows developers to analyze and develop software for the HI model and create a space in which software engineering and ML experts can work together on the ML model life-cycle, especially in the health field. This manuscript proposed a framework that included a theoretical framework composed of four modules (software, ML model, ML algorithms, and HI data). The new methodology was compared between three system engineering methods: Vee, Agile and SEMLHI. The results showed the delivery of the new methodology for one-shot delivery. For the MAM component on the SEMLHI framework, laboratory test results were obtained using five algorithms to test the accuracy of the ICD-10 results using equations and to evaluate the accuracy of the ML models with a sample size of 750 patients. The results for MAM showed that the SVG was approximately 0.57.

9. REFERENCES

[1] A. Holzinger, ''Interactive machine learning: Experimental evidence for the human in the algorithmic loop,'' Appl. Intell., vol. 49, no. 7, pp. 2401–2414, 2019.

[2] T. A. Mohammed, A. Ghareeb, H. Al-Bayaty, and S. Aljawarneh, ''Big data challenges and achievements: Applications on smart cities and energy sector,'' in Proc. 2nd Int. Conf. Data Sci., E-Learn. Inf. Syst., 2019, p. 26.

[3] B. Cakici, K. Hebing, M. Grünewald, P. Saretok, and A. Hulth, ''CASE: A framework for computer supported outbreak detection,'' BMC Med. Inform. Decis. Making, vol. 10, no. 1, p. 14, 2010.

[4] A. J. Vickers, T. Salz, E. Basch, M. R. Cooperberg, P. R. Carroll, F. Tighe, and J. Eastham, and R. C. Rosen, ''Electronic patient self-assessment and management (SAM): A novel framework for cancer survivorship,'' BMC Med. Inform. Decis. Making, vol. 10, no. 1, p. 34, 2010.

[5] A. Ismail, A. Shehab, and I. M. El-Henawy, ''Healthcare analysis in smart big data analytics: Reviews, challenges and recommendations,'' in Security in Smart Cities: Models, Applications, and Challenges, vol. 9, A. E. Hassanien, M. Elhoseny, S. H. Ahmed, and A. K. Singh, Eds. Cham, Switzerland: Springer, Nov. 2019, pp. 27–45.

[6] J. F. Bobb, B. C. Henn, L. Valeri, and B. A. Coull, ''Statistical software for analyzing the health effects of multiple concurrent exposures via Bayesian kernel machine regression,'' Environ. Health, vol. 17, no. 1, p. 67, 2018.

[7] B. Aribisala and O. Olabanjo, ''Medical image processor and repository– MIPAR,'' Inform. Med. Unlocked, vol. 12, pp. 75–80, Jul. 2018.

[8] W. Aigner and S. Miksch, ''CareVis: Integrated visualization of computerized protocols and temporal patient data,'' Artif. Intell.in Med., vol. 37, no. 3, pp. 203–218, Jul. 2006.

[9] J. Krause, A. Perer, and H. Stavropoulos, ''Supporting iterative cohort construction with visual temporal queries,''

IEEE Trans. Vis. Comput. Graph., vol. 22, no. 1, pp. 91–100, Jan. 2016.
[10] R. K. Pathinarupothi, P. Durga, and E. S. Rangan, ‘‘Data to diagnosis in global health: A 3P approach,’’ BMC Med. Inform. Decis. Making, vol. 18, no. 1, pp. 1–13, 2018.
[11] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, Data Mining: Practical Machine Learning Tools and Techniques. Amsterdam, The Netherlands: Elsevier, 2016, pp. 438–441.
[12] Q.-C. To, J. Soto, and V. Markl, ‘‘A survey of state management in big data processing systems,’’ VLDB J., vol. 27, no. 6, pp. 847–872, Dec. 2018. [13] S. R. Salkuti, ‘‘A survey of big data and machine learning,’’ Int. J. Elect. Comput. Eng., to be published. Accessed: Jan. 7, 2020. [Online]. Available: http://ijece.iaescore.com/index.php/IJECE/article/view/19184/pdf
[14] F. Khomh, B. Adams, J. Cheng, M. Fokaefs, and G. Antoniol, ‘‘Software engineering for machine-learning applications: The road ahead,’’ IEEE Softw., vol. 35, no. 5, pp. 81–84, Sep. 2018.
[15] T. A. Mohammed, Y. I. Hamodi, and N. T. Yousir, ‘‘Intelligent enhancement of organization work flow and work scheduling using machine learning approach tree algorithm,’’ Int. J. Comput. Sci. Netw. Secur., vol. 18, no. 6, pp. 87–90, 2018.
[16] J. A. Diao, I. S. Kohane, and A. K. Manrai, ‘‘Biomedical informatics and machine learning for clinical genomics,’’ Hum. Mol. Genet., vol. 27, no. R1, pp. R29–R34, May 2018.
[17] P.-H. Cheng, Y.-P. Chen, and J.-S. Lai, ‘‘An interflow system requirement analysis in health informatics field,’’ in Proc. WRI World Congr. Comput. Sci. Inf. Eng., vol. 1, 2009, pp. 712–716.
[18] C. George, P. Duquenoy, and D. Whitehouse, ‘‘eHealth: Legal, ethical and governance challenges,’’ in eHealth: Legal, Ethical and Governance Challenges, C. George, D. Whitehouse, and P. Duquenoy, Eds. Berlin, Germany: Springer, 2014, pp. 1–398.
[19] K. N. Mishra and C. Chakraborty, ‘‘A novel approach towards using big data and IoT for improving the efficiency of m-health systems,’’ in Advanced Computational Intelligence Techniques for Virtual Reality in Healthcare, vol. 875. Cham, Switzerland: Springer, 2020, pp. 123–139.
[20] B. Farahani, M. Barzegari, F. Shams Aliee, and K. A. Shaik, ‘‘Towards collaborative intelligent IoT eHealth: From device to fog, and cloud,’’ Microprocessors Microsyst., vol. 72, Feb. 2020, Art. no. 102938.