International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2020 IJSRCSEIT | Volume 6 | Issue 3 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT206371

A Survey on Risk Analysis in Information Technology Infrastructure Library (ITIL) Change Management Using Supervised Machine Learning

Srushti Gajjar, Mrugendrasinh Rahevar
U. & PU. Patel Department of Computer Engineering, Chandubhai S. Patel Institute of Technology, CSPIT, Charotar University of Science and Technology, Charusat, Changa, India

ABSTRACT

Innovation in IT and technology leads to new developments within the organization. It is important for companies to respond more quickly to the changing trends in order to stay competitive. ITIL change management allows companies to introduce new technologies without interruption or downtime. It follows a standard practice to avoid any unwanted interruptions and involves evaluation, planning and approval of changes. Change Management is all about managing risk for the company and it is linked to the perception of risk that the company has. Risk Analysis is primary component when it comes to any software changes; organizations are concerned about risk management. For better performance by identifying and assessing risk in systematic manner is the aim of the risk management. In ITIL change management risk assessment is a manual process. Automation of risk analysis would have enormous benefits, like reducing the downtime, maximize the productivity and so on. So this paper is mainly on the survey of different supervised machine algorithms of machine learning, like support vector machine, Naive Bayes, logistic regressions.

Keywords : ITIL Change Management, Risk Analysis, Support Vector Machine, Naïve Bayes, Logistic Regression, KNN

I. INTRODUCTION

Company today's need a framework to manage their global change governance a maintaining some processes to comply with government regulation and keeping customer's satisfaction. IT groups receive a large kind of requests from customers together with access to applications, computer code enhancements, laptop upgrades and new mobile phones. The Information Technology Infrastructure Library (ITIL) classifies these varieties of requests as a 'request for service' and identifies the method to handle service requests as 'request fulfillment several service requests are continual, therefore to realize the best potency, a repeatable method and procedure ought to be outlined. Change management is that formal service transition for emerging technology smoothly and efficiently changes the technology. Change management to be there to make it smoother and more efficient that technology being corporate within the business. With each iteration of technology there is a chance of may be misalignments infrastructure as a whole, meaning it may have flaws or bugs that need to workout. For these reasons many companies decide to set the new technologies in parallel with each other.

With new technologies being adopted and user requests increasing, business requirements are constantly changing. IT environment is necessary for precisely handling and controlling the above changes. Therefore, IT is regarded as a crucial factor for business. Many problems can lead to Changes, but

many Changes are proactively used for business purposes as it reduces costs and improves services. "To reduce the impact change related incidents of any organization, for that systematic procedures are used for efficient handling of all changes," is aim of the Change Management process. Scenarios where ITIL Change Management is used and Risk analysis process is critical: New data centre implementation, Window's patch, OS Update, Restoring of database for developer, etc. Automation of Risk Analysis would have enormous benefits. Reduce in downtime, Increase in optimal productivity, Proper acquisition of tools and resources, etc.

One of the phase, which needs improvement is the risk assessment process is manually in ITIL Change Management. Risk Assessment process is error prone, time consuming and costly and technical uncertainty when it is done manually. Software projects have more expensive and complexity, so it has more risks than other management projects. Automation of risk analysis would have enormous benefits like reduce in downtime, increase in optimal productivity, proper acquisition of tools and resources, etc. This paper is mainly focusing on survey of machine learning classification algorithms for risk analysis.

Critical projects or key business initiatives are highly impacted by potential issues either in negative or positive way, the procedure of analyzing these issues is considered to be as Risk analysis. There are various notions of risk in areas of Safety System, finance, industrial activities, Technologies etc and continuous analysis of risk analysis is being carried out in these areas[1]. A project operation should be considered "risky" if: the probability of a bad outcome is strong, the ability to influence it within the project's time and resources limits is poor and its potential consequences are serious [2]. Project risk management (PRM) is essential for project performance and is indispensable. The risks in projects have actually increased in terms of number and global effects. Projects are more transparent and risk-averse than ever, and stakeholders are asking for more risk management to mitigate financial or legal implications. That is why it has become increasingly important to manage project risks effectively and efficiently, in order to give project stakeholders a higher guarantee of success and comfort, or at least to warn them of potential problems or disasters [3].

II. Risks in Software Projects

Some mistakes are done in software development projects but which are not noticeable and it is responsible to harm the software, which is called software project risks. And even if after identifying the risk factor if we ignore it, it will maximize the possibility of failures of any project. Before it harms to any software, it is mandatory to minimize the risk [4]. Some critical factors like time, budget, resources, etc are influenced by risk factors [5]. There are more risks in software projects than other projects because of software projects are more complex and also have technical unreliability. The success or failure of any project influenced by risk factor, so for that efficient method need to apply for reduce the risks for better management [6]. Also the reasons for software project failure are inefficient and inadequate risk management [7]. The proper risk management increases the probabilities of positive impact minimize probability of negative impact [6]. The convenient risk management is reason for success of the project. Risk management activity as follow the process as identify the risks, analyze, planning, Tracking, Control, Communication. So, the proper risk management is responsible for minimize impact on the project and different Supervised Learning mechanisms are adopted by researchers to analyze the risks. Prompt action to reduce the possibility and/or impact of a risk that occurs on the project is often more successful than attempting to recover after the risk occurs.

Table 1 : Failures of different projects and loss (Cost in US $) [8]

Year	Company	Outcome (Cost in US $)
2005	Hudson Bay Co. [Canada]	Inventory system issues add $33.3 million * in loss
2004-05	UK Inland Revenue	Software errors lead to tax-credit overpayment of $3.45 billion *
2004	Avis Europe PLC [UK]	Enterprise Resource Planning system scrapped after investing $54.5 million
2003-04	AT & T Wireless	Problems with CRM updates led to $100 in revenue loss
2002	CIGNA Corp.	CRM program issues led to $445 million loss
1999	Hershey Foods Corp.	Problems with the ERP system contributed to a loss of $151million.
1996	Arianespace [France]	Software specification and design errors causing Ariane 5 rocket to crash at $350 million.
1996	FoxMeyer Drug Co.	$40 million ERP program discontinued after deployment causing bankruptcy of business
1994	Chemical Bank	Software error creates deductions of a total of $15 million from 100,000 customer accounts.

III. Background

Many researchers have done or doing researches on risk analysis using different models and methods. They were used different machine learning algorithms like support vector machine (SVM), Naïve Bayes (NB), Multiple Logistic Regression, etc.

Table 2 : Research work in Risk Analysis

Paper	Year	Techniques
Prediction of risk factors of software development project by using multiple logistic regression [6].	2015	Factor Analysis and Multiple Logistic Regression
Classification of Software Project Risk Factors Using Machine Learning Approach [4].	2016	Analytical Hierarchy Process (AHP) and Support Vector Machine
Designing a Machine Learning Based Software Risk Assessment Model Using Naïve Bayes Algorithm [5].	2018	Naïve Bayes Classification
Towards an efficient risk assessment in software projects– Fuzzy reinforcement paradigm [9].	2017	Fuzzy Multi-Criteria Decision Making

A Probabilistic Software Risk Assessment and Estimation Model for Software Projects [10].	2015	Bayesian Belief Network

A. *Multiple logistic Regressions:*

The research approach [6] is according to predict the risks in software projects using multiple logistic regressions which can control the software development process. It also contains the risk factor analysis and stratum. In this factor analysis method also used and it is integrated with logistic regression. Classification of risk is predicted by above scenario for possibility of success or failure of software projects. One factor relates to another through factor analysis. Less relevant factor or not related factor are not dependent upon any direction like positive, negative or contra. The independent factors combine with continuous factor on the basis of multiple logistic regression analysis as it works on the basis of grouping and predicting on those dichotomous factors such as YES and NO. The separation of all factors into risk and non-risk groups done by multiple logistic regression using static method. The above analysis results compare the result of predicted risks which were analyzed in to Odds which means we can be arranged the risk to compared between them. Collecting the Indicators of the risk factor used for estimation of the risk. The software project analyzed, studied and categorized into low, medium and high-risk factors. A summary of the indicators have been selected from the genetics which are classified into software risks. The experts will help to determine the optimized questionnaires. The questionnaires related to risk factors were reliable as they were selected by expertise. In this, data are collected from the experts who had given the similar judgment and accord, probably accord and disagree. Grouping and separation of the factors as to quality analysis is termed as method of factor extraction. There is importance of result of questionnaire which was answered by involved personnel. At the end multiple logistic regression is used for analyzes the results of the respondents. This primary classification of risk is mainly focusing on the risks that are again and again happen in software projects. Some risk factors by this research are User, Software Requirements, Estimations, Cost, Schedule, Planning and Control, etc.

B. *Support Vector Machine:*

Researchers in paper [4] are using other machine learning methods as support vector machine which is popular machine learning algorithm for regression, classification and for others. Here Machine Learning approach is used because it is model which depends on the changing risk factors. Considerable upgrades of the modern performing approaches are successfully done by Support Vector Machine and they are automatic. Support Vector Machines (SVMs) construct a decision surface in the feature space that bisects the two categories and maximizes the margin of separation between two classes of points.

(To build a widen margin of hyperplane, in order to isolate into one class of data to another is the main focus of the SVM.) In this research Analytical Hierarchical Process(AHP) is used to give priority to risk factors and at the same scalability issues are obtained by using SVM. Combination of impact and possibility of risk factors are used to give the priority to the risk.

In this research work, proposed approach follows the four steps: The first step is Risk Factor Elicitation where the experts like the project managers and risk owners who are directly involved in the risk management process and interviewing them. Based on their organizational experience, those participants were given a questionnaire. Risk factors of the software project are established by expert judgment [3].The second step is Pair Wise Comparison among the risk factors. The third step is Risk Classification

where Support Vector Machine is used. SVM constructs a decision boundary [11][12] between the class of hard bound risk factors and the class of less hard bound risk factors and maximizes the margin of separation of two classes. The fourth step involves checking for the change in requirements. Check for change in requirements after classifying the risk factors and visualizing the results. If the software project requirements have changed, then again elicit the risk factors and repeat step 1 to step 3 in order to achieve the scalability factor. So finally used machine learning approach provides the accurate result but the SVM input dataset is done manually, so if risk factors increase, then the effort required by human evaluators is rapidly increasing.

C. Naïve Bayes:

Researchers in [5] proposed a Supervised Learning mechanism and its Naive Bayesian (NB) Classification. To evaluate the parameters is the nature of naive Bayesian classification. In complicated real-world situations there is better performance by it. According to this paper, the first step is Input Selection where dataset is as a input. The next step is Data Pre-processing which is for dataset cleaning. After the data pre-processing next step is Classification which is categorizing the records. Naive Bayes (NB) Classification algorithm is used in implementation phase. In supervised learning setting, Naïve Bayes classifier is trained efficiently. Naïve Bayes algorithm predicts the critical risk and the parameter of the process calculate the performance. Depend on high, low, mid and normal risk, the graph is evaluated and probability values estimate these in dataset. In this paper comparison of different machine learning techniques are given, where the Naïve Bayes has the better accuracy compared to others.

D. Fuzzy Reinforce Learning:

In research paper [14], researchers are analyzing Independent Assessments that are carried out very early in the life cycle of software development. They have developed the Research Prototype Early Assessment System, which is fuzzy expert system used for Independent Project Evaluation during the early stages of the life cycle of the project. Fuzzy logic provides an easy way to view subjective probability, linguistic variables and ordinary categories. Subjective likelihood is higher than quantitative empirical likelihood of failure to represent a risk. They showed how the use of the minimal information about a current project and software development guidelines will derive useful results from fuzzy expert systems.

To efficiently identify and rank the notable risks of the software projects there is hybridizing fuzzy multi criteria decision making approaches for the development of an assessment framework is used. That will help for decision making during software product's life cycle. In this specific research work, they have integrated the Fuzzy Multi-Criteria Decision Making and TODIM approaches for assessment of project risk. The methodology proposed was used to provide an efficient rating method for measuring risk factors in software. The usefulness of the proposed results of the methodology was contrasted with earlier approaches for software project risk assessment [9].

One research work is on software risk indicators for risk assessment where research work is proposes a probabilistic software risk estimation model using Bayesian Belief Network (BNN). The data collected from software projects used by organization performs an empirical experiment to assess the constructed model. The used model is differing from extant models where extant model does not consider the software risk indicator's uncertainty [10].

IV. CONCLUSION

In this survey, studied the different machine learning approaches for software projects such as support vector machine, naïve bayes, multiple logistic regression, fuzzy reinforce learning, etc. Because of the manual risk analysis process in ITIL change management, it is very time-consuming, prone to error, less effective and expensive so that we can

automate the risk analysis process in ITIL change management specifically for software changes using different machine leaning approaches. According to various researcher's reasoning good risk management increases the probability of positive impact minimizes the probability of negative impact. Efficient risk management is responsible for reducing project impact, and researchers follow different Supervised Learning Frameworks for risk analysis.

V. REFERENCES

[1]. M. Goman, "Towards unambiguous IT risk definition," in Proceedings of the Central European Cybersecurity Conference 2018 on - CECC 2018, Ljubljana, Slovenia, 2018, pp. 1–6, doi: 10.1145/3277570.3277586.

[2]. J. A. Keizera, J. I. M. Halman, and M. Song, "From experience: applying the risk diagnosing methodology," J Product Innovation Man, vol. 19, no. 3, pp. 213–232, May 2002, doi: 10.1111/1540-5885.1930213.

[3]. C. Fang and F. Marle, "A simulation-based risk network model for decision support in project risk management," Decision Support Systems, vol. 52, no. 3, pp. 635–644, Feb. 2012, doi: 10.1016/j.dss.2011.10.021.

[4]. P. Chaudhary, D. Singh, and A. Sharma, "Classification of Software Project Risk Factors Using Machine Learning Approach," in Intelligent Systems Technologies and Applications, vol. 385, S. Berretti, S. M. Thampi, and S. Dasgupta, Eds. Cham: Springer International Publishing, 2016, pp. 297–309.

[5]. S. K and D. R, "Designing a Machine Learning Based Software Risk Assessment Model Using Naïve Bayes Algorithm," TAGA JOURNAL, vol. VOL. 14, pp. 3141–3147, 2018.

[6]. T. Christiansen, P. Wuttidittachotti, S. Prakancharoen, and S. A. Vallipakorn, "Prediction of Risk Factors of Software Development Project By Using Multiple Regression," Asian Research Publishing Network (ARPN), vol. VOL. 10, pp. 1–9, Feb. 2015.

[7]. Y. Hu, X. Zhang, E. W. T. Ngai, R. Cai, and M. Liu, "Software project risk analysis using Bayesian networks with causality constraints," Decision Support Systems, vol. 56, pp. 439–449, Dec. 2013, doi: 10.1016/j.dss.2012.11.001.

[8]. "Why Software Fails." IEEE Spectrum, Sep-2005.

[9]. A. K. Sangaiah, O. W. Samuel, X. Li, M. Abdel-Basset, and H. Wang, "Towards an efficient risk assessment in software projects–Fuzzy reinforcement paradigm," Computers & Electrical Engineering, vol. 71, pp. 833–846, Oct. 2018, doi: 10.1016/j.compeleceng.2017.07.022.

[10]. C. Kumar and D. K. Yadav, "A Probabilistic Software Risk Assessment and Estimation Model for Software Projects," Procedia Computer Science, vol. 54, pp. 353–361, 2015, doi: 10.1016/j.procs.2015.06.041.

[11]. J. C. B. CHRISTOPHER, "A Tutorial on Support Vector Machines for Pattern Recognition." Kluwer Academic Publishers, Boston, 1998.

[12]. J. S.N., "Efficient Classification Algorithms using SVMs for Large Datasets," Supercomputer Education and Research Center INDIAN INSTITUTE OF SCIENCE BANGALORE – 560 012, INDIA, Jun. 2007.

[13]. Z. Xu, T. M. Khoshgoftaar, and E. B. Allen, "Application of fuzzy expert systems in assessing operational risk of software," Information and Software Technology, vol. 45, no. 7, pp. 373–388, May 2003, doi: 10.1016/S0950-5849(03)00010-7.

Cite this article as :

Srushti Gajjar, Mrugendrasinh Rahevar , "A Survey on Risk Analysis in Information Technology Infrastructure Library (ITIL) Change Management Using Supervised Machine Learning ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6, Issue 3, pp.298-303, May-June-2020. Available at doi : https://doi.org/10.32628/CSEIT206371
Journal URL : http://ijsrcseit.com/CSEIT206371