Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 8, July 2021: 4566-4576

Agile practices in global software development: A topic modelling approach

Prem Prakash Dayal[a], Dr. M Rajkumar[b]

[a] Research Scholar, ICFAI University, Jharkhand
[b] Associate Professor, ICFAI University, Jharkhand

Abstract

Agile practices have made a significant impact in Global Software Development (GSD). Literature review papers on agile practices in global software development have been limited by constraints of traditional methods to manually select and analyse research papers. The purpose of this study is to leverage an enhanced automated way to analyse literature to establish dimensions of agile global software development covering around 144 journal articles in SCOPUS. The study unravels key themes latent in the academic literature, identify dominant topics and summarise interpretation around agile practices deployed in global software development setup. This study uses an effective and efficient topic modelling by using LDA method and identifies four dominant clusters.

Keywords: Agile Practices, Global Software Development, Topic Modelling, LDA

1. Introduction

Globalisation has caused an increase in global software development in the IT industry. Global Software Development (GSD) or Global Software Engineering (GSE) or Distributed Software Development (DSD) can be defined as the activity of developing the software with teams spread across different geographic locations, organizations, nationalities, culture, and time zones (Kaur & Sharma, 2014).GSD has become increasingly popular among IT organizations as it offers savings in development cost, provides access to large pool of skilled labour and reduced time to market (Conchúir et al., 2006). While GSD has advantages, it can also pose several challenges such as communication, collaboration, and coordination due to temporal, geographical, cultural, and linguistic distances(Zahedi et al., 2016).While adoption of GSD poses coordination, collaboration, and communication challenges (Carmel, 2001). However several practices have been suggested to overcome these challenges. Regular client visits, synchronous communication technology and knowledge sharing platforms are few practices that Noll et al., (2010) have discussed to overcome GSD challenges. Additionally, one such practice that has managed to mitigate global distance challenges in GSD has been the adoption of agile software development practices (Holmstrm et al., 2006).

The manifesto comprising of four values and twelve principles for Agile Software Development (ASD) was published in 2001. Beck et al., (2001) had outlined the core values of the manifesto: individuals and interactions over processes and tools; working software over comprehensive documentation; customer collaboration over contract negotiation; and responding to change over following a plan.

Interestingly, most of the agile methodologies (e.g. scrum) assume that the team is co-located. Unfortunately, this principle does not fit in the real scenario where agile teams are also distributed across the globe. As per the recent ((14th Annual State of Agile report published by Digital.ai, 2020) there is an increasing trend for team collaboration across geographic boundaries and time zones and 81% of respondents said their organization has Agile teams where the members of the same team do not all work in the same location (i.e. not co-located).Hence, it is all the more imperative to study about the agile practices in global software development setup.

While a considerable amount of manual effort gets spent to manually review large amount of research papers however with advancement in natural language processing, methods such as topic modelling becomes less demanding on human subjects(Kee et al., 2019).While systematic literature reviews on agile in distributed software development or global software development have been carried out by authors (Vallon et al., 2018) there exists no reviews carried out using the topic modelling approach. Unlike the manual process of analysing literature which is time consuming, the topic modelling approach is an efficient and systematic approach that involves the usage of computer algorithms to find out latent topics of word occurrence in large document collections. Topic models involves building a collection of topics for the corpus and a collection of topic proportions for each of its documents(Chang et al., 2009). The topic modelling approach is also being used to efficiently perform literature reviews also known as smart literature reviews of research papers(Asmussen & Møller, 2019; Karami et al., 2020; Kingdom et al., 2019).

This paper is an attempt to carry out a smart literature review using topic modelling on research papers that study benefits and challenges in distributed agile global software development environment.

2. Theoretical Background

Global Software Development

An unprecedented rise in globalization trends had witnessed a paradigm shift in the software industry. Software development had become globally distributed, multisite, and multicultural (Herbsleb & Moitra, 2001).Several factors have conspired to bring a change from centralised software development to globally distributed software development. The pressure to reduce lead time to market by adopting round the clock development, to seek cost and location advantage in global resourcing , and the need to gain market proximity advantage for the business are key forces contributing to global software development(Herbsleb & Moitra, 2001).

Global Software Development (GSD) phenomenon which began in the 1990s is characterized by geographically distributed project teams. Unlike the centralised software development where members have rich collaboration , communication and control(Ågerfalk et al., 2005), adoption of GSD introduces collaboration challenges among distributed project teams. GSD teams need to collaborate over geographic, temporal, linguistic and cultural distances collectively called “global distance” (Noll et al., 2010).

Agile

According to Alsahli et al., (2017) agile software development is a philosophy and not a single methodology or a single set of tools. The ASD Philosophy developed in 2001 is strongly rooted in four

values and 12 principles. ASD values individuals and interactions over processes and tools, working software over comprehensive documentation, Customer collaboration over contract negotiation and responding to change over following a plan. Continuous delivery as opposed to final delivery of the complete product; face to face conversation as opposed to heavy documentation; accepting change in requirements even late as opposed to defining fixed set of requirements (Fowler & Highsmith, 2001)

Agile software development methods (example: Extreme Programming,SCRUM,etc) comprise of agile practices such as daily stand-ups , pair programming etc are taken up by teams in their routines of software development(Gupta et al., 2019).

Agile Global Software Development

While working together in a co-located manner is desirable for agile practices however currently there is no trend of increase in co-location. Instead, post covid-19 pandemic, distributed agile practices are being predominantly followed with the rise of distributed agile teams (Digital.ai, 2020).

Hence, as summarised by Jalali & Wohlin, (2012) even though agile methods are best suitable with co-located customers and developers who have frequent interactions, software organizations have reported the incorporation of agile in GSD. However, besides benefits there are extensive challenges associated and a lot of effort is required to let the agile in GSD work. Challenges associated with communication, collaboration, culture, knowledge management etc have been reported and organizations have reported solutions and tactics to mitigate these challenges.

Literature review through Topic Modelling

Literature review on a topic is conducted through various methods. Widely used methods that are followed are Systematic literature review etc where the researcher searches the article manually using topics found as keywords. After reading the contents, the recording of the data is carried out and the gaps are identified.

Unlike manual literature review, the smart literature review is an alternative in recent times which uses a topic modelling approach (Asmussen & Møller, 2019) by which a budding researcher can review large documents within a short time period by identifying hidden themes, from the existing literature.

Smart lit review has been used by researchers through topic modelling(Asmussen & Møller, 2019; Karami et al., 2020). Topic modelling has been used in various areas such as content analysis(Hagen, 2018) , twitter analysis (Mutanga & Abayomi, 2020)

Topic modelling is a natural language processing (NLP) method used to uncover the hidden thematic structures in underlying document collections. There are several algorithms that are used to perform topic modelling for example Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA) etc. As an example, topic models are used to browse abstracts of large collections of articles without relying on metadata (Dantu et al., 2021; Eickhoff & Neuss, 2017). Topic Models also known as probabilistic topic models is an unsupervised analysis of large document collections. Each document is described as a mixture of topics. With algorithms like LDA we can use topic models to discover both the topics and an assignment of topics to documents from a collection of documents(Chang et al., 2009). The algorithm chosen for the purpose of this study is LDA. LDA is an unsupervised algorithm that

automatically creates multiple topics based on the patterns of co-occurrence of words in each document being analysed (Jacobi et al., 2016).

3. How Topic Modelling works

Topic modelling is a Natural Language Processing (NLP) method used to uncover the hidden thematic structures in underlying document collections. There are several algorithms that are used to perform topic modelling for example Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA) etc. As an example, topic models are used to browse abstracts of large collections of articles without relying on metadata (Dantu et al., 2021; Eickhoff & Neuss, 2017). Topic Models also known as probabilistic topic models is an unsupervised analysis of large document collections. Each document is described as a mixture of topics. With algorithms like LDA we can use topic models to discover both the topics and an assignment of topics to documents from a collection of documents(Chang et al., 2009). The algorithm chosen for the purpose of this study is LDA. LDA is an unsupervised algorithm that automatically creates multiple topics based on the patterns of co-occurrence of words in each document being analysed (Jacobi et al., 2016).

Literatures were also traced on conducting topic modelling in the research work made by Kee et al., (2019) and Dantu et al., (2021)

4. Method

The three steps namely , pre-processing , topic modelling and post processing outlined in a framework given by Asmussen & Møller, (2019) has been followed as part of smart literature review using topic modelling . LDA was chosen as the algorithm to identify the latent themes in the abstracts. Topic modelling using LDA was done largely in three steps, by creating a framework as pre- processing, topic modelling and post processing.

5. Data Collection

The abstract of a journal article provides succinct information about the overall details contained in the paper. Hence, we extracted abstracts of the research articles as per the search below string from the SCOPUS databased.

The below search used was divided into three parts. The first part contains all the relevant keywords related to agile. The second part of the string uses the relevant keywords about global software development. The third part of the search string contains keywords related to benefits and challenges.

Part 1

("agile" OR "scrum" OR "extreme programming" OR "pair programming" OR "lean development" OR "lean software development" OR "Kanban" OR "XP" OR "ASD" OR "Agile Project Management" OR "Test-Driven Development" OR "Agility" OR "Feature Driven Development" OR "Crystal Family" OR "Dynamic System development Methodology" OR "TDD" OR "FDD" OR "DSDM" OR "ASD" OR "Adaptive Software Development" OR "Rational Unified Process" OR "RUP")

Part 2

AND ("global software engineering" OR "global software development" OR "distributed software engineering" OR "distributed software development" OR "GSE" OR "GSD" OR "distributed team" OR "global team" OR "dispersed team" OR "spread team" OR "virtual team" OR "offshore" OR "outsource" OR "Global Outsourcing" OR "Multi-site development")

Part 3

AND ("Challenges" OR "hurdles" OR "Factors" OR "Difficulties" OR "Parameters" OR "Motivators" OR "Variables" OR "Characteristics" OR "Barriers" OR "Reasons" OR "enablers" OR "Motivation" OR "Advantage" OR "Constraints" OR "Benefits" OR "Scepticism" OR "Incentive" OR " OR " issues " OR " value " OR " determinants "") AND (LIMIT-TO (DOCTYPE , "ar")) AND (LIMIT-TO (LANGUAGE , "English")) AND (LIMIT-TO (SRCTYPE , "j")) .

Step1: Pre-processing

As part of the pre-processing step, all the 144 journal articles' abstracts were loaded. Post downloading the 144 abstracts, manual review led to cleaning up of 54 articles. Hence, 90 was the final count of abstract used for further steps. The csv file containing the cleaned up 90 abstracts was uploaded. Subsequently, all the non-value adding words were removed from the loaded 90 abstracts. After, converting all the words to lower case, the punctuations and whitespaces were removed. Special characters stop words like "can"," make" and other non-semantic contributing words were removed as part of this pre-processing step. Additionally, words were stemmed to their root form for easier comparison.

The decision around the number of topics was an important activity. For, the purpose of clarity, four dominant topics were considered for the study.

Step2: Topic modelling

Once the abstracts were cleaned and the decision was made on the number of topics, the LDA algorithm was then executed. The outcome of the LDA model was a list of articles, a list of probabilities for each document for each topic, and a list of most frequent words for each topic.

Step3: Post- processing

As part of the post-processing step, a list of topic probabilities for each article was generated as an outcome of the LDA model. The topics were then identified and labelled. The researcher then identified the main topic in each topic group and labelled each topic. To avoid subjective bias of the researcher, a combination of reviewing the most frequent words for each topic and the title review was utilised.

Semantic validation was used as a validation process in the proposed framework. At first, the title of the individual research paper was reviewed to validate that every article did belong to its respective topic. As LDA is an unsupervised method, it was assumed that not all papers will have a perfect fit within each topic, however if most papers happened to be within the theme of the topic, it was considered as a valid result

6. Results

Topic Cluster 1: Practice

This cluster includes research more around the agile practices. This cluster discusses topics related to risk identification, assessment and mitigation of GSD projects using agile (Rafeek et al., 2019). There have been discussions around challenges in managing distributed project teams spread across different geographic boundaries and how flexible management approach could help mitigate this challenge (Yadav, 2016).Additionally, authors (such as Beecham et al., 2020; Esteki et al., 2020; Shrivastava & Rathod, 2017) also talk about distributed software project risks and risk management frameworks to mitigate such as agile scaling frameworks (SAFe) , scrum using PRINCE2 etc. Studies such as Lee & Yong, (2010) has also focussed on project management challenges of distributed agile projects with reference to project quality and productivity. Studies such as (Al-Zaidi & Qureshi, 2017; Dalcher & Raffo, 2009) focus on improving the effectiveness of agile practices such as SCRUM, distributed pair programming (Satratzemi et al., 2018) ; Savage, 1952) and so on in mitigating global software development challenges. Knowledge management is an important aspect and studies such as (Dalcher & Raffo, 2009) focus on how practices to bridge knowledge boundaries among team members help avoid project delays. As part of this cluster, keywords such as practice, scrum, method, approach, management, risk, knowledge are found associated. This cluster forms the biggest of four clusters.

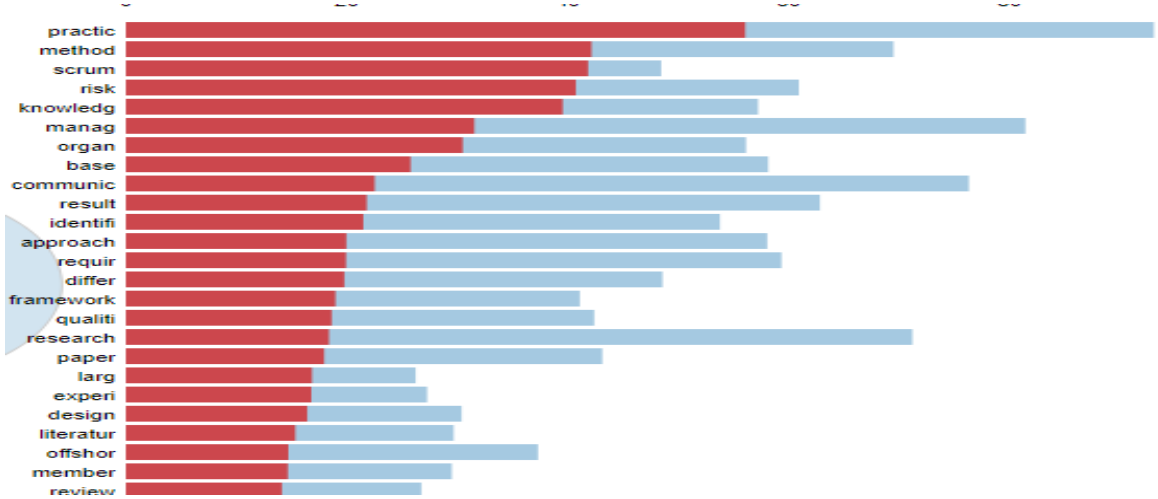

Figure 1

Topic Cluster 2: Process

This cluster includes research related to benefits and challenges around process in agile global software development. This cluster discusses on topic around challenges in knowledge sharing and process to knowledge management(Borrego et al., 2017; Sungkur & Ramasawmy, 2014).Additionally, authors have also discussed on topics relating to tailoring of agile development processes to overcome team coordination and communication challenges(Bass, 2016; Estler et al., 2014; Jain & Suman, 2016). One of the highly discussed topics in this cluster is knowledge management challenges. One important observation in this cluster is about lack of theory- based studies. In this cluster, we observed only grounded theory used. With keywords such as process, factor, model, method and so on , this cluster forms the second largest chunk of all four.

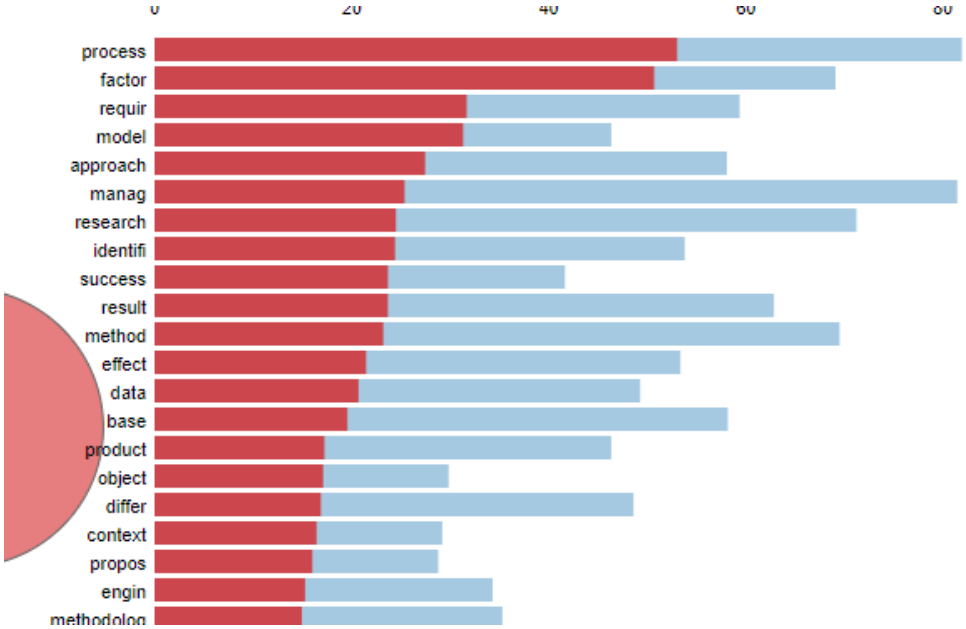

Figure 2

Topic Cluster 3: Technology

This cluster includes research related to benefits and challenges around tools and technology around agile global software development. Topics around leveraging tools to overcome communication challenges in agile global software development is discussed (Yagüe et al., 2016).Moreover, authors such as Gill, (2015) discuss about a social technology tool called communication technology assessment tool and provide evidence of how it proves to improve the effectiveness in communication and collaboration around agile global software development context. Additionally, as the usage global or distributed teams scale, authors like Dhar, (2012) mention how cloud computing helps benefit organization agile in their agile adoption and authors such as Belsis et al., (2014) discuss around the usage of intelligent processing techniques help improve effectiveness of requirement engineering. Important keywords as part of this cluster are technology, collaboration, communication and so on.

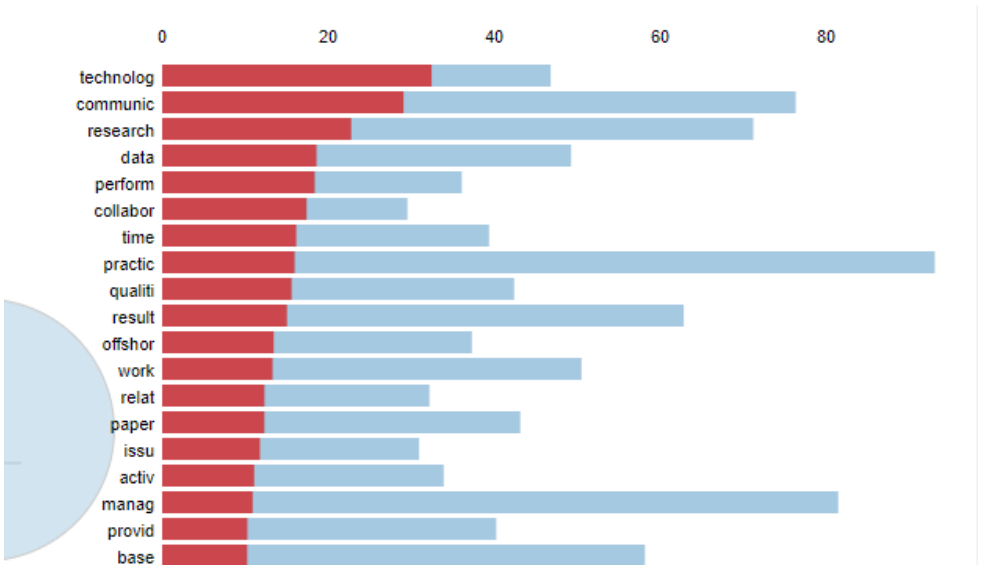

Figure 3

Topic Cluster 4: Communication

This cluster includes research around the benefits and challenges around the communication in agile global software development. Authors (Luz et al., 2009; Shameem et al., 2018) discuss how effective communication plays an important role in the effectiveness of agility in distributed setup. Additionally , other than challenges, there are studies such as Layman et al., (2006) where informal communication practices play as critical enabling factor in the distributed software development success. This cluster happens to be the smallest of the four. Notable keywords are communication, framework , management, product and so on.

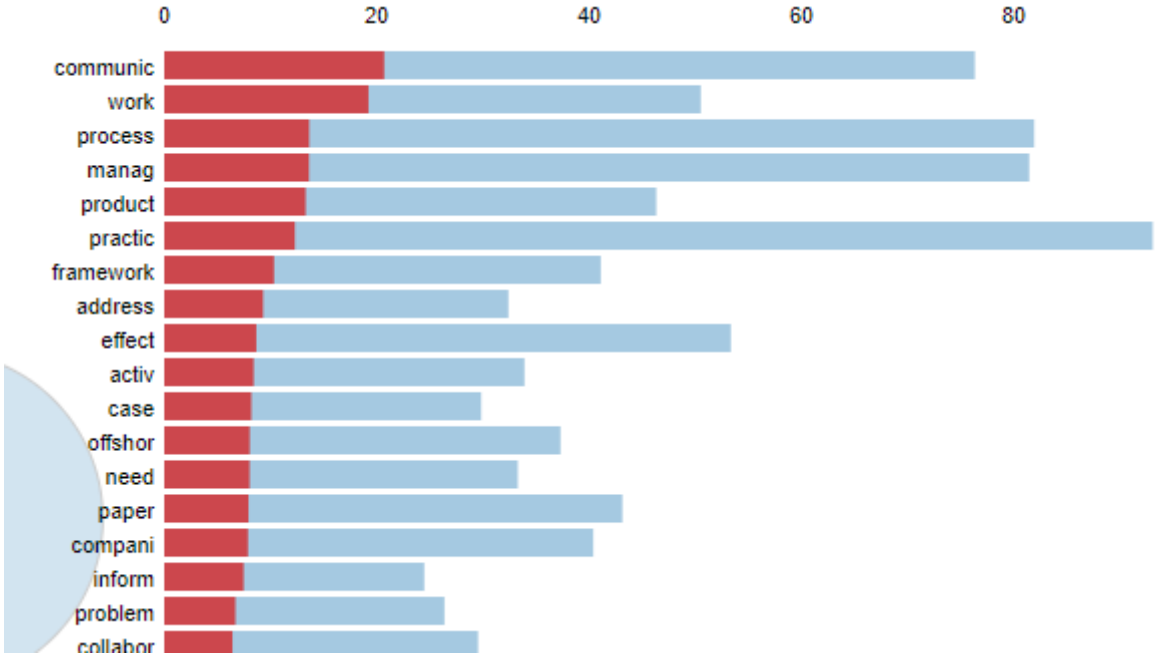

Figure 4

6. Discussion

The approach of using topic modelling for literature review is not common. The framework proposed by Asmussen & Møller, (2020) was used in this paper in order to leverage topic modelling for the literature review of 144 papers on agile global software development. The framework helped save lot of manual effort. While the framework considered the entire document, this study has considered only abstracts of the research articles.

This study unearths key themes that have been largely discussed by scholars around the benefits and challenges experienced when agile and GSD are combined. It implies that global agile teams in general and organizations can leverage value from agile in GSD only if they focus on overcoming challenges that arise. As part of this study, four broad clusters were finalised to categorise the themes. While, the ideas within each cluster could overlap however each cluster has a unique focus. The first refers to discussions around benefits and challenges from the practice per se. The second focuses more around the process and tools. The third and fourth clusters focus on technology and communication, respectively. The adoption of Topic Modelling approach has helped researchers save a lot of manual effort.

7. Conclusion and Future Research

Like any other research, this research too has some limitations. The number of papers considered is 144 which is relatively less. However, enough efforts were made by the researcher in identifying and extracting the abstracts using various search strings. The current study helps provide an insight into various research streams being studied by research scholars in the agile global software development. However, it does not allow us to understand how research in academia compares with state-of -the-art in the industry.

We propose that future research might focus on white papers from industry to identify additional research challenges and opportunities in the agile global software development.

In the current study we have focussed on categorising benefits and challenges in four broad clusters. The focus predominantly has been on identifying what all themes have researchers been studying w.r.t benefits and challenges when agile and global software development are combined. However, deeper analysis of topics using topic modelling has not been undertaken. Though the current study has limitation, this is certainly an important step to understand what all research has been undertaken to understand different kinds of challenges when agile practices are combined with global software development. Additionally, this study can help give directions to researchers to extend the realms of knowledge about agile global software development. As part of this study only the abstracts and not the entire text of the 90 odd articles was used for topic modelling. As a future study, we recommend researchers to take up more and more theory-based studies to examine challenges in agile global software development. We also suggest if researchers perform studies around understanding the challenges in agile GSD through the lens of behavioural theories. Lastly, we also recommend researchers to study motivation at the team level in the context agile and GSD.

References

[1]. Ågerfalk, P. J., Fitzgerald, B., Holmström, H., Lings, B., Lundell, B., & Conchúir, E. Ó. (2005). A framework for considering opportunities and threats in distributed software development. Proc. International Workshop on Distributed Software Development, Paris, France: Austrian Computer Society, August, 47–61.

[2]. Al-Zaidi, A., & Qureshi, R. (2017). Global software development geographical distance communication challenges. International Arab Journal of Information Technology, 14(2), 215–222.

[3]. Alsahli, A., Khan, H., & Alyahya, S. (2017). Agile Development Overcomes GSD Challenges: A Systematic Review. International Journal of Computer Science and Software Engineering, 6(1).

[4]. Asmussen, C. B., & Møller, C. (2019). Smart literature review: a practical topic modelling approach to exploratory literature review. Journal of Big Data, 6(1).

[5]. Asmussen, C. B., & Møller, C. (2020). Enabling supply chain analytics for enterprise information systems: a topic modelling literature review and future research agenda. Enterprise Information Systems, 14(5), 563–610.

[6]. Bass, J. M. (2016). Artefacts and agile method tailoring in large-scale offshore software development programmes. Information and Software Technology, 75, 1–16. h

[7]. Beck, K., Grenning, J., Martin, R., Beedle, M., Highsmith, J., & Mellor, S. (2001). Manifesto for Agile Software Development. The Agile Alliance, 12–14.

[8]. Beecham, S., Clear, T., Lal, R., & Noll, J. (2020). Do scaling agile frameworks address global software development risks? An empirical study. ArXiv.

[9]. Belsis, P., Koutoumanos, A., & Sgouropoulou, C. (2014). PBURC: A patterns-based, unsupervised requirements clustering framework for distributed agile software development. Requirements Engineering, 19(2), 213–225.

[10]. Borrego, G., Morán, A. L., Cinco, R. R. P., Rodríguez-Elias, O. M., & García-Canseco, E. (2017). Review of approaches to manage architectural knowledge in Agile Global Software Development. IET Software, 11(3), 77–

88.

[11]. Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. Advances in Neural Information Processing Systems 22 - Proceedings of the 2009 Conference, 288–296.

[12]. Conchúir, E. Ó., Holmström, H., Ågerfalk, P. J., & Fitzgerald, B. (2006). Benefits of Global Software Development , International Conference on Global Software Engineering Exploring the Assumed Benefits of Global Software Development Development Global.

[13]. Dalcher, D., & Raffo, D. (2009). Software Process : The End of an Era. Practice, November, 303–304.

[14]. Dantu, R., Dissanayake, I., & Nerur, S. (2021). Exploratory Analysis of Internet of Things (IoT) in Healthcare: A Topic Modelling & Co-citation Approaches. Information Systems Management, 38(1), 62–78.

[15]. Dhar, S. (2012). From outsourcing to Cloud computing: Evolution of IT services. Management Research Review, 35(8), 664–675. https://doi.org/10.1108/01409171211247677Digital.ai. (2020). 14th annual STATE OF AGILE REPORT. Annual Report for the STATE OF AGILE, 14(14), 2–19.

[16]. Eickhoff, M., & Neuss, N. (2017). Topic Modelling Methodology. Proceedings of the 25th European Conference on Information Systems (ECIS), 2017, 1327–1347.

[17]. Erran Carmel, R. A. (2001). Global Software Development Tactical Approaches for Alleviating Distance in Global Software Development. IEEE SOFTWARE.

[18]. Esteki, M., Gandomani, T. J., & Farsani, H. K. (2020). A risk management framework for distributed scrum using prince2 methodology. Bulletin of Electrical Engineering and Informatics, 9(3), 1299–1310.

[19]. Estler, H. C., Nordio, M., Furia, C. A., Meyer, B., & Schneider, J. (2014). Agile vs. structured distributed software development: A case study. Empirical Software Engineering, 19(5), 1197–1224.

[20]. Gill, A. Q. (2015). Distributed Agile development: Applying a coverage analysis approach to the evaluation of a communication technology assessment tool. International Journal of E-Collaboration, 11(1), 57–76.

[21]. Gupta, M., George, J. F., & Xia, W. (2019). Relationships between IT department culture and agile software development practices: An empirical investigation. International Journal of Information Management, 44(September 2018), 13–24.

[22]. Hagen, L. (2018). Content analysis of e-petitions with topic modeling: How to train and evaluate LDA models? Information Processing & Management.

[23]. Herbsleb, J. D., & Moitra, D. (2001). Global software development. IEEE Software, 18(2), 16–20.

[24]. Holmstrm, H., Fitzgerald, B., Agerfalk, P. J., & Conchoir, E. (2006). Contemporary Practices in Systems Development Agile Practices Reduce Distance in Global Software Development. Information Systems Management, 7–19.

[25]. Jacobi, C., Van Atteveldt, W., & Welbers, K. (2016). Quantitative analysis of large amounts of journalistic texts using topic modelling. Digital Journalism, 4(1), 89–106.

[26]. Jain, R., & Suman, U. (2016). Effectiveness of agile practices in global software development. International Journal of Grid and Distributed Computing, 9(10), 231–248.

[27]. Jalali, S., & Wohlin, C. (2012). Global software engineering and agile practices: A systematic review. Journal of Software: Evolution and Process, 24(6), 643–659.

[28]. Karami, A., Lundy, M., Webb, F., & Dwivedi, Y. K. (2020). Twitter and Research: A Systematic Literature Review through Text Mining. IEEE Access, 8, 67698–67717.

[29]. Kaur, P., & Sharma, S. (2014). Agile Software Development in Global Software Engineering. International Journal of Computer Applications, 97(4), 39–43.

[30]. Kee, Y. H., Li, C., Kong, L. C., Tang, C. J., & Chuang, K. L. (2019). Scoping Review of Mindfulness Research: a Topic Modelling Approach. Mindfulness, 10(8), 1474–1488.

[31]. Kingdom, U., Science, M., Kingdom, U., & Science, M. (2019). Repositório ISCTE-IUL. Meat Science, 102,(351), 49–58.

[32]. Layman, L., Williams, L., Damian, D., & Bures, H. (2006). Essential communication practices for Extreme Programming in a global software development team. Information and Software Technology, 48(9), 781–794.

[33]. Lee, S., & Yong, H. S. (2010). Distributed agile: Project management in a global environment. Empirical Software Engineering, 15(2), 204–217.

[34]. Luz, M., Gazineu, D., & Teófilo, M. (2009). Challenges on adopting scrum for distributed teams in home office environments. World Academy of Science, Engineering and Technology, 59(December 2013), 308–311.

[35]. Mutanga, M. B., & Abayomi, A. (2020). Tweeting on COVID-19 pandemic in South Africa: LDA-based topic modelling approach. African Journal of Science, Technology, Innovation and Development, 0(0), 1–10.

[36]. Noll, J., Beecham, S., & Richardson, I. (2010). Global software development and collaboration: Barriers and solutions. ACM Inroads, 1(3), 66–78.

[37]. Rafeek, M. A., Arbain, A. F., & Sudarmilah, E. (2019). Risk mitigation techniques in agile development processes. International Journal of Supply Chain Management, 8(2), 1123–1129.

[38]. Satratzemi, M., Xinogalos, S., Tsompanoudi, D., & Karamitopoulos, L. (2018). Examining Student Performance and Attitudes on Distributed Pair Programming. Scientific Programming, 2018.

[39]. Savage, J. M. (1952). of the Academy of Sciences. 52(December).

[40]. Shameem, M., Kumar, R. R., Kumar, C., Chandra, B., & Khan, A. A. (2018). Prioritizing challenges of agile process in distributed software development environment using analytic hierarchy process. Journal of Software: Evolution and Process, 30(11).

[41]. Shrivastava, S. V., & Rathod, U. (2017). A risk management framework for distributed agile projects. Information and Software Technology, 85(October 2020), 1–15.

[42]. Sungkur, R. K., & Ramasawmy, M. (2014). Knowledge4Scrum, a novel knowledge management tool for agile distributed teams. Vine, 44(3), 394–419.

[43]. Vallon, R., da Silva Estácio, B. J., Prikladnicki, R., & Grechenig, T. (2018). Systematic literature review on agile practices in global software development. Information and Software Technology, 96(December 2017), 161–180.

[44]. Yadav, V. (2016). A Flexible Management Approach for Globally Distributed Software Projects. Global Journal of Flexible Systems Management, 17(1), 29–40.

[45]. Yagüe, A., Garbajosa, J., Díaz, J., & González, E. (2016). An exploratory study in communication in Agile Global Software Development. Computer Standards and Interfaces, 48(September), 184–197.

[46]. Zahedi, M., Shahin, M., & Ali Babar, M. (2016). A systematic review of knowledge sharing challenges and practices in global software development. International Journal of Information Management, 36(6), 995–1019.