# SOFTWARE ENGINEERING AND ARTIFICIAL INTELLIGENCE: RE-ENHANCING THE LIFECYCLE

Sabeer Saeed[1] and Asaf Varol[2]

[1]Department of Software Engineering, Firat University, Elazig City, Turkey
[2]Department of Computer Engineering,
Maltepe University, Istanbul City, Turkey

***ABSTRACT***

*As automation is changing everything in today's world, there is an urgent need for artificial intelligence, the basic component of today's automation and innovation to have standards for software engineering for analysis and design before it is synthesized to avoid disaster. Artificial intelligence software can make development costs and time easier for programmers. There is a probability that society may reject artificial intelligence unless a trustworthy standard in software engineering is created to make them safe. For society to have more confidence in artificial intelligence applications or systems, researchers and practitioners in computing industry need to work not only on the cross-section of artificial intelligence and software engineering, but also on software theory that can serve as a universal framework for software development, most especially in artificial intelligence systems. This paper seeks to(a) encourage the development of standards in artificial intelligence that will immensely contribute to the development of software engineering industry considering the fact that artificial intelligence is one of the leading technologies driving innovation worldwide (b) Propose the need for professional bodies from philosophy, law, medicine, engineering, government, international community (such as NATO, UN), and science and technology bodies to develop a standardized framework on how AI can work in the future that can guarantee safety to the public among others. These standards will boost public confidence and guarantee acceptance of artificial intelligence applications or systems by both the end-users and the general public.*

***KEYWORDS***

*Software Engineering (SE); Artificial Intelligence (AI); Machine Learning (ML); Deep Learning (DL); Cross Section between Artificial Intelligence and Software Engineering.*

## 1. INTRODUCTION

Today's world activities are driven by technology. Technology is an indispensable tool as it affects the way we work, the way we play, and the way we live generally in today's society. Technology is the ability that human beings have with tools to sustain our environment and the world at large. Decades after decades, several new technologies emerge. The invention of the integrated circuit chip by Jack Kilby, the work of Bardeen and Walter House Brattain has brought about what is called transistor which was among the reasons for personal computers [1].

As a result, individuals can acquire a computer system, which led to high demand for software which offers software developers the myth like 'We already have standby framework that is full

David C. Wyld et al. (Eds): CSE, AI & FL, NLPTT - 2021
pp. 33-44, 2021. CS & IT - CSCP 2021 DOI: 10.5121/csit.2021.112403

of principles, guidelines, and approaches for developing software. Why would my people 'profile' again after having the entirety of all the required know-how to do?'[2]. Instead of without concern whether: those frameworks or standards can be applied or alive? Software professionals are mindful of the framework? Does it mirror the present or latest software engineering profession and implementation? Is it conclusive? Is it favourable to organizations? Is it more efficient to increase delivery-time while also continuing to work on effectiveness? In most situations, the response to all these problems is 'no,' those were among the events that could not be answered decades back that was called 'software crises' [2].

As computers, physical systems (hardware) results are, therefore, subject to the laws of physics, which are what eventually influence what they can or cannot do, at least in theory [1]. The introduction of engineering concepts to software development (which give birth to Software Engineering (SE)) by NATO to solve the software crises was a great achievement that today automating business is done not just done successfully, but profitably. Product (i.e. software) is software-engineered that is nearly different with respect to malleability [3].

Despite the progress made in software engineering, some researchers argued 'is yet to clearly give formal methods to software design and development', perhaps, which AI might assist in conventionalization of the software design and development [3]. Artificial intelligence (AI) keeps rising from giving the ability to computers to act based on the knowledge rules set to it, to Machine Learning where a computer is trained to learn in order to perform given tasks.

'During the 1950s when the researchers were working on the new area called AI in computer science with the purpose of 'can computer think like human beings?' or can computers perform a function like human beings?' They accepted the fact that computers can perform functions that can be discharged by humans [4]. Machine learning (ML) is the process of training a computer to learn from experience in n set of tasks. Deep learning (DL) is an advanced ML as AI encompasses both DL and ML. The value of AI is worthwhile to research to see how influential it is to SE while avoiding the pitfall. For AI to be acceptable, both engineers, lawyers, philosophers, environmental experts, and the government need to set a clear regulation. The most researched areas in AI are concerned with its abilities for mathematics and algorithm that made it capable to understand, perform, and improve from experience [5]. While the research in SE deals with enabling human beings with easiness 'tools' or 'process' to create a software that meets the need of business; to perform the task efficiently and effectively; to be easy to operate, use and maintain; and to accept modification change and do those changing for better and future enhancement [2].

Traditional software follows rigorous stages of software development life cycle from requirement elicitation to testing and implementation but AI mostly is an environmental trial. Both fields are impacting each other in the world of technology by giving tremendous progress to humanity. The study will review works to see how they can contribute to each other.
The study able to:

- ⇒ Outline how Artificial Intelligence can contribute more to Software Engineering.
- ⇒ Highlight how Software Engineering can contribute more to Artificial Intelligence.
- ⇒ Emphasize to stakeholders in the SE and AI sector not just to work on remaking the safety of autonomous vehicle (AI) but also to work on software theory for all technology such as AI that requires software to operate
- ⇒ Proposed for the need for professionals' bodies from philosophy, law, doctors, engineers, government, international communities (such as NATO, UN) and, science and technology communities to develop a professional approach, set up a high standard (on

how AI can work in the future that can guarantee safety)and be committed to the public interest.

## 2. ARTIFICIAL INTELLIGENCE

AI emergence begins in the 1950s, as a concept that gives the ability for computers to act like humans. It was expressed by John McCarthy (Lisp programming language pioneer [7]). McCarthy and his colleagues' work on AI can be stated in four words: Intelligent performance requires knowledge [7]. In 1843, when lady Ada Lovelace made a statement about her colleague, Charles Babbage's work on 'Analytical Engine,' she said, the manifestation of the 'engine was to assist us, humans, in doing what we know 'how' but quicker. That is, the machine can only do anything we know how to command it to do.' So, she rejected the idea of artificial intelligence. But she was contradicted by Alan Turing (AI pioneer) on his paper titled 'Computing Machinery and Intelligence' during the 1950s.

The AI pioneer revealed the Turing test as well as key technicalities that would come to entangle as AI [4]. The Pioneer, after going over the Ada Lovelace quotes while considering the alternative general-purpose-machine could be able to learn and think independently, and he came to the conclusion that they could [4]. AI investigators attain to believe intelligence by pleasing suitably to deliver the cause of intelligence as an intelligent act. Among the principle of AI procedure and collection is that advancement is 'sought' by developing a system that does 'synthesis' before 'analysis' [5].

But building the AI system by synthesizing before analysis rounding up to start operation may have the chance to end up like that of Amazon when they decided to develop an AI system that would help them 'filter CVs of potential candidates. The AI was selecting some set of CVs while also rejecting some. The 'Algorithm was tested to be biased' against females. They found out that, the reason for the algorithm rejecting women was due to 'certain traits' known to 'netball' in the hobbies segment. So, the training set was unfair which led to the scrapping of the project [8].

Nevertheless, AI is bringing more types of knowledge to witness. This means it reflects changes concurrently because, AI, of last decades, is not the AI of today. To mention a few AI is continuing to make impacts in healthcare services, email spam management, natural language processing (NLP), speech recognition, etc. In SE, AI is transforming natural language requirement (NL) into design and specification that can be used for description and data types in programming [9].

Also, in software requirements [9], the author develops a class-model builder (CM Builder) that helps in building a class diagram specified in Unified Modeling Language (UML) from the NL requirement document. AI Knowledge Base System (KBS) includes three oriented layers: knowledge acquisitive layer (which can be human-familiarized, sensible, informal), the depiction layer (convention, logical), and the implementation layer (machine-familiarized, data structures, and systems) are used to capture design relatives upon the development of the needed input and output of the system activity [7, 9].

Researchers also initiated the Ontology-Based Software Development Environment (ODE) from the software process existence [9]. The concept of Computational Intelligence (CI) techniques was built to help and encourage requirements engineering through intelligence computing. AI concepts for software architecture development, the Robyn Lutz as cited in [9], applied Genetic Algorithms (GAs) to seek the interval of potential ranking 'decompositions of a system.' She also further her investigation on Product Line Architectures (PLA) to place different points are clearly

and obviously defined to improve 'reusability and editability' of mention architecture that likely to be applied to exemplify the class set of architectures.

Experts system can help programmers in coding and testing programming process. But automate programming process could be applied by using AI techniques such as analogical reasoning to software reuse [9]. Equivalence of analogical reasoning is Case-Based Reasoning (CBR) that solves similar problems with similar solutions. Another is Experience Factory (EF), also, for reuse and management of experience, knowledge, process, and product. EF is likewise popularly known as the Learning Software Organization (LSO). There are reusable techniques such as Knowledge Acquisition (KA) and Domain Modeling (DM).

Another AI knack is Constraint programming that is applied in software engineering. For example, it is expanded to program the PTIDEJ system (Pattern Trace Identification, Detection, and Enhancement in Java), an automated system created to recognize micro-architectures mirror-like design form models in object-oriented source code. A research field appear to emphasis on expressing viewpoints of Software Engineering as problems that may be answered using meta-heuristic search algorithms built-in AI is Search-Based Software Engineering (SBSE). This SBSE is the reestablishment of software engineering function as an optimization to solving problems. Such examples of the optimization and search abilities that can be applied are genetic algorithms [9].

As for testing**,** still remains an expensive function in the Software development life cycle. AI researchers have been committed to the application of constraint solving concepts in the automation of program testing (Constraint-based testing). CBT refers to as the method of generating test scenarios or cases from software or models by applying the constraint programming technology. Handling up-to or more than hundreds of thousands of lines of code, with vigorous built structure particularly like enormous dynamic data structures, in the company of non-sequential numerical constraints obtained from ambiguous statements or facts are some of the issues we have to face with. Expandability is the major question that CBT tools have to answer for us [9].

Such AI techniques can distinguish parallel features with many dimensions, including the scope of the purpose to address? What power level of automation? What portion level of the course from not formal requirements to ML is mechanized? Purpose of what portions of the system lifecycle is addressed? Knowledge of what types of know-how information is used by the machine?

AI capabilities that have been tested to be serving a purpose in SE research and implementation can be referred to as 'Probabilistic SE', 'Classification, Learning, and Prediction for SE', and 'Search-Base SE.' Figure 1 Showing AI and its sub-field.

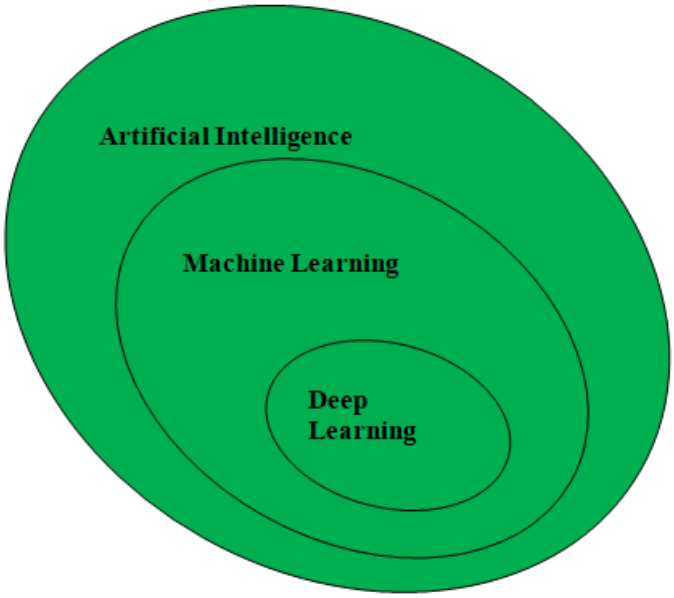


Figure. 1. adopted from [4]. Showing AI and its intra areas

### 2.1. Why AI Needs Research From SE?

In order to standardize AI, we need research from SE to systematically support the production of AI systems and applications, the operations in the real ecosystem, together with examining, maintaining, and modifying to increase more features and for future enhancement. And, to the discovery of the best among the better alternatives in terms of scientific & systematic approaches and methods so that we can have conventional models and architectures for AI application systems.

SE domain can enable the description of AI application methods and can serve as an information dictionary to all elements used in the development process [7]. These can make an easy way in the future to have a universal software theory that could support the development of AI or other technology so that it will give room for the legal framework in case if there is for breach of contract (or warranty) between customers and manufacturers/developers.

## 3. SOFTWARE ENGINEERING

Software Engineering it the systematic approach, application of mathematical technique, and the scientific method for developing, operating, maintaining, enhancing, and retiring software at an affordable cost to the customer. The discipline was introduced in the 1960s by NATO following the 'software crises.' The study found out that as a result of innovations of microchips transistors and integrated circuits that led to the affordability of personal computers to individuals, which in turn led to high demand for software. Also, there was inadequate standard to software development, and programmers at that time thought that '*After we develop the software and set it to operate, our task is finish' or 'the solely deployable function service for an achievement software project is the functioning software.'* While in real existence, the functioning software is merely one segment of a software arrangement or setup that brings many components. A specific type of service products (e.g., models, documents, plans) support for building successful engineering, more substantial, and control for software support.

[2] Cited that 'A software philosopher once stated that earlier beginning to developing software code, then, the extension of the period to complete it done. Also, research from industry information discovered that from 60 to 80 percentages of all tasks expended on software expect to be spent after it is deployed to the client for the first time' [2]. The NATO discussion at Garmisch-Partenkirchen, Germany, the Major aim is to have an efficient and effective production way of high-standard and mostly huge software systems. The objective is to help and encourage software engineers and managers to develop credible software in a quicker way with tools, methods, discipline, and processes by adopting engineering techniques such as:

- **Tools**. Which can be automated or semi-automated enablers for software processes and methods.
- **Method**. The procedural technical 'know-how' for developing software.
- **Process**. The 'glue' between that technological layers together and support rational and development to the software.

Therefore, SE is the standard development of software not theory. There are many other arguments for the use of theories. Moreover, 'theories provide common frameworks that allow the structuring of knowledge in a precise and concise path, that facilitates the communication of ideas and knowledge independent space and time. 'Nonetheless, the usefulness of theories for software development is at this moment a discussion issue and the current use of theories in this discipline is not well known' [1]. Even thou Simon presents that 'more and more, computers will program themselves' during a time of the great expectation for AI, this expectation did not come to fruition at that time [11].

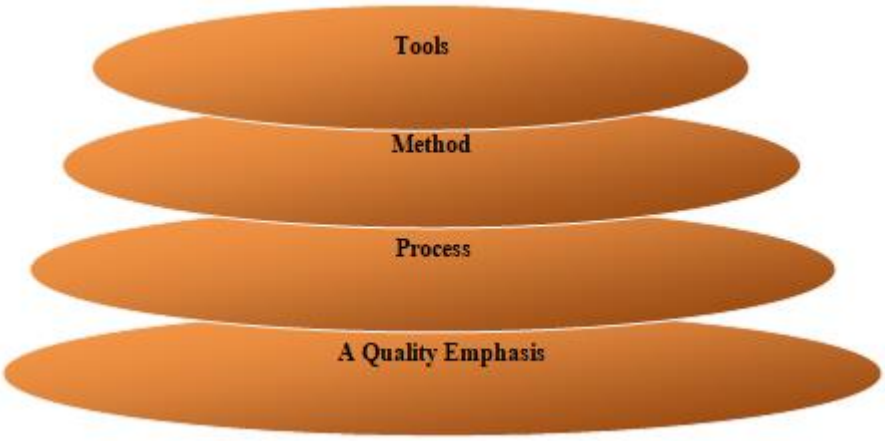


Figure 2. Showing software engineering standards structured [2]

## 3.1. Why SE need Research from AI?

Why SE aimed at research from AI? As we are growing into the internet-of-things (IoT), AI can make the inclusion of those technologies more productive as well as to serve as an ecosystem for testing software and models. This research can enhance the civic-legal framework to raise up user acceptance. SE methods may be re-engineered using AI concepts, which can lead to perceivably robust technology. And, can help in the preparation of a data dictionary for describing the meaning of the respective SE methods and tools. In annex, other new AI techniques that are extending or trending in a virtual environment can be applied for a better variation of the SE approaches [7].

## 4. CROSS-SECTION BETWEEN SE AND AI

As the cross-section of AI and SE is presently uncommon, but increasingly are getting higher. Nowadays, pathways and methods from both major encourage research and implementation relative to each other research area. Figure 3 below from [5] represent some research fields in AI and SE sector as well as their cross-section.

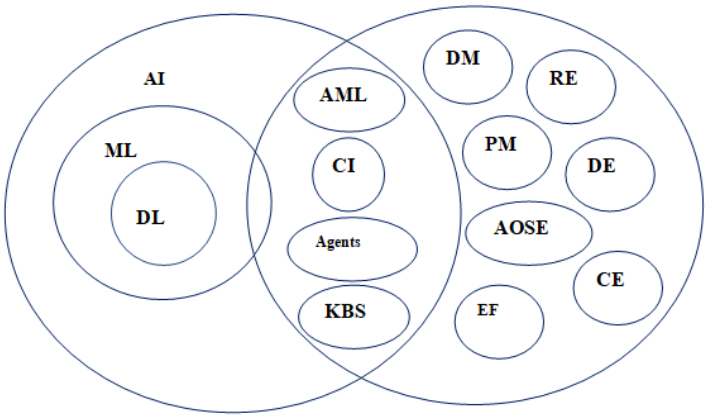


Figure 3. Showing research fields in AI and SE and their inter-sections [5].

AI system or application is similar to many programs and applications; merely more, therefore, the building of AI software can be figured as a progressive form of ideal maintenance often needed to transform specification. Traditional SE procedure produce both cohesion and adhesion, function unwell in such frequent specification changing condition, so we need to create more SE possibilities techniques that can enable rapid application development and evolution [7].

We apply the concept of SE to the development of production project management (P-PM), creation of requirement engineering (RE), designing engineering (DE), software source code engineering (CE), and method models to enable the development of an intelligent system as well as using advanced data analysis. KA advancement can simplify to makes EF and reasoning atmosphere system like DM advancement which enables the development of models for generating the requirement for system programs and services product lines [5].

Between CBR and EF, the former is applied for accessing an information to the latter. SE applied the concept of information agent to model (or replicate) production processes or to share and show substitute requests [5].

The sophistication of AI is getting more value, that is worth vital to examine how it is of assistance to SE while preventing possible drawbacks. AI development which can lead to a high sum of preferences, reasoning, and autonomy path to crucial law and ethical issues whose solution affects both experts and customers of AI technology [6]. These issues cline between civil code of conduct, rules of law, public matters, the practitioner ethics, philosophical policy, and the likes that will require intelligentsias from AI scientists (or Engineers), software engineers, legal practitioners, political analysts, and advocators to answer [6].

Even nowadays a customer is limited to sue for any damage to breach-of-contract in commercial-off-the-shelf software, in fact, most cases end up discouraging the customer due to inadequate professional standards in the industry. So how significant are we trying to ensure those sets of principles are put in place for customers to have confidence in the system for AI?

Vladeck brought an example in the year 2014, that, if the autonomous driving vehicle is deployed in a country like the US to reduce approximately 40,000 yearly traffic inevitabilities and crashes in one of two equal figures, then, the manufacturers could get more than 20,000 damages lawsuits instead of thousand feedback messages of appreciation notes from the customers. But the question is 'in what legal regulation or legislation capacity can guarantee the certainty of the autonomous self-driving car or drone flying vehicle to be best possibly achieved?' [6].

Because if not so, that can be repudiated due to silence by the eyes of the law of the customer right. Therefore, shall the legal issues concerning AI to be managed or regulated by the current software & internet cyberlaw, or should they be managed differently? (Calo 2014b) cited in [6]. Differently in the sense that, should we enact new laws and create a new regulator for it? But with the profession of SE, enacting the law and creating the code of conduct bureau can be easier especially in the future if we are able to have global software system theory.

Some researchers proposed that society will ignore autonomously technologies except there are some trustworthy ways of propelling them to wherewithal safe. SE for AI can help to ascertain the indefinite areas in which an AI may inadequately act as desired equivalent to various fields of robustness research. An example of such SE concept to successful design and implementation of software is verification and validation.

## 4.1. Verification

Verification is the process of confirming in order to substantiate that system is able to act what it formally requested to perform. Did we develop the system correctly? Verification measure mean approaches that assure a highly reliable set of formal conditions that AI software must certify to give room for a legal framework and the likes. If so, then, the persuasion for AI and other automation technologies to scale critical safety measures and to reach satisfactory state-of-affairs so that they can be acceptable to the society will be victorious.

### 4.1.1. Existing Standard on Verification: Its Relevance to AI System

Previously, the standard verification of software accelerated significantly. A notable example of that is the work of Klein and his colleagues in the year 2009, on a fully common purpose operating kernel named as SeL4kernel. Klein and his colleague mathematically examined the device base on the standardized specification to assert an assurance against dangerous operations, mortality, and crashes. The verification technique and approach methods are to set up high guarantee software tools (Fisher 2012) [6]. With the help of SE, we should not just only develop an AI system that can be verified on the underlying layer, but rather we should also verify the design of the AI system especially, if the designs follow a componentized architecture, in such a way it guarantees that every single part-contained component can be integrated according to the functions and connections to reflect the attributes of the main system [6].

### 4.1.2. Verification Difference Between SE and AI

Research shows that approximately when developing a technological device, the noticeable difference between SE and AI in terms of verification is the outcome of SE which is software is implemented with the thoughtfulness of a known mathematical and machine model, while in contrast to the AI systems particularly robot that function in the ecosystem that is relatively known by the system experts. In today's turbulent environment, SE can come up with a model to give AI full support for universal acceptance due to convention verification.

### 4.2. Validation

The is the state of being authentic. Validation is to measure the system to see how it corresponds to its formal requirement by not having unacceptable features with repercussions. Did we develop the correct system? Validity measures: is to affirm and ground unwanted attributes that may exist/happen regardless of the system is formally proper. In the future terms, research shows that AI systems influentially can have more capabilities and self-control, in such instance unjustifiable validity could progress to huge damages. Healthy and wealthy assurance for machine-learning methods, a field 'represented as for not-long-term validity research, will also be significant for long-term validity for safety' [6].

### 4.3. Other Requirement and Support for Creation of Legal Framework for AI

SE verification and validation can only allow for the development of AI whether autonomously (or not) to an absolute correctness level. But in case of if there could be 'hidden' unrecovered 'defect' that can only be detected when the customer has almost spent more than a month using it. That defect could cause economic damages, crashes, and/or fatalities. How can a customer seek restitution of such damages? Vendors of COTS software is only with 'hard to convince' to be liable in the view of the law for fixing any fault, defects, and failure no matter how substantial the damage is as far as COTS license and law are concerned. Even if the vendor can be proven of the negligence of duty in the development process customer can only be difficult to claim his/her right because of a lack of a strong professional body that certifies the software engineering profession.

Also, lack of professional government agencies in some countries that regulate and standardize software related issues. But had it been it is in the building sector where there are building and construction regulators that regulate and develop the necessary standards governing all the construction affairs. They make sure the building meets the required features and standards while building practitioners must be certified by the building professional organ. The building professional body evaluates the civil engineers, architectures, quantity surveyors, regional planners to list a few, in order to make sure that individual does not claim to be knowledgeable about the field but has acquired the technical knowledge of know-how about the field. So, for a vendor or developer of an AI to be able to be responsible for breach of warranty or contract then there is a need for such a system of the construction sector to be initiated in AI or ICT in general.
We need to have a universal SE and AI professional body that certifies who wants to pursue a career in the sector so that we can have a chartered software engineer or chartered artificial intelligence engineer and so on. To have an SE and AI regulatory commission that will regulate the activities of the technology, and also, to have legislation or law that can protect the interest of both customers and engineers. For the law to be acceptable in society there is a need for collaboration from the legislators, advocators, lawyers, philosophers, political analysts, psychologists, media practitioners, academicians, innovators, entrepreneurs, engineers, scientists, and the likes.

## 5. RECOMMENDATIONS

The study encourages that the future of AI can be ascertained with the help of SE as SE already relevantly benefitted from AI. Let make an example of how the world came to accomplish automobile car into existence in the society in the US. The revolution of the automobile car was a success to the individuals, economics, and change to countries landscape at large. But it also has it on ups-and-down at its inception stage.

Around the first decades of the twentieth century, the automobile was recognized with the rise of fatal and accidental death and injuries that suggested an expression of great concern. A community to dialogize the issues was inaugurated which involves safety advocates, engineers, physicians, media practitioners, and others related to contribute their various opinion about the causes of the fatalities, injuries, and accidents. They revealed up that what causes these tragedies was from the driving ethics, design of automobiles, to environmental hazards and highway road engineering.

So, to minimize those tragic consequences and to retain the precedence of automobile effort began from regulating driver character, redesigning of the automobile, to the reformation of highway road and environment. The automobile took almost or more than a decade to recognize, prioritize, and regulate these risk agents.

Thus, this study recommends a similar approach to avoid such tragedies and gain public confidence in adopting AI in society. As humankind, we should try to get the possible best of AI before deploying it to our ecosystem. AI should or must be at least 99.98% accurate that can give assurance to the people. All stakeholders from the industry, academia, government, research agencies, and other related should able to answer questions like 'can this AI (like an autonomous vehicle) be adopted in any environment whether rural or urban?' 'Does it require more sophistication knowledge from the user be he/she can use it?' Or 'anybody with or without literacy can use it? 'What navigation system does it use? Is it a radioactive signal? If it is a radioactive signal then what will happen to the product supposing if it is limited or unavailable?
As the comprehensive testing of AI sometimes is complex and difficult to put into practice or time-consuming application of automation using simulation can be helpful. One vital aspect of SE that can help to accomplish quality assurance to provide guaranteed AI products to the public is verification and validation (V&V). Once if the stakeholders can ascertain AI quality assurance according to the ecosystem and public character or behavior then the deployment of AI can begin getting people's attention and confidence.

Although these days there has been an increase in research grants on software theory which can give of a phenomenal guarantee of software life spam maybe even on Autonomous AI i.e. if the research reach its peak. The study hopes so for the impact of software theory especially on AI can reduced some resources spending and give a platform for universal AI where every citizen can have a right from confidence level to a legal aspect in case if he or she is not satisfied.

## 6. CONCLUSIONS

As the technique of AI to SE makes an impact achievement to SE, then we need to work more on SE for AI. Especially the concept of verification and validation from the techniques of SE can make the autonomous vehicle easy to implement to the market as it can give AI quality assurance to the public. As people expect the ideal factor AI to be a perfect autonomous vehicle that varied from what it is in reality the study cross-check between AI and SE and find out the possible feature of each segment that can be solution to boost each other.

For us to be able to create an autonomous AI that can work robustly accurate, then all AI communities must of course need to come up with good attributes that can be applied to every domain. And the standard issues need to also find the solution to questions like what SE formal abilities are currently available? How reliable are they? What kind of research should undertake to find more abilities and concept? And what concept can we interchange such that all fields of computer science and broader AI expertise can be made useful worthy? To exemplify this is the work of Wallach and Allen (2008) that concern for substantial relevant factor is the logical value of a variety set of formal standards (or ethical code of conduct): we may need to apply little

worth of approximations if the formal standard cannot be organized well enough to influence behaviour in the safety-critical state of affairs.

'There is nothing so practical as a good theory' [1]. theories explain one specific aspect of reality, according to what is known from it up to the present time, and they enable predicting future events [1]. Today's software architecture is largely described as 'emulate to adopt', so there is no accepted software architecture widely, not to talk of software architecture for AI. So, researchers need to do more in between the SE and AI to reduce AI development time, effort, and cost. Also, to have an acceptance SE model to AI in such a way that is not just only SE standard to AI, but conventional software theory and software architecture that can standardize the development of AI and other technologies.

## ACKNOWLEDGEMENTS

We will like to thank the Almighty God for this priceless opportunity called life, because with life then there is hope contribute to human progress through academics.

## REFERENCES

[1] Andrade, J. Ares, García, R. Martínez, M. Rodríguez, S. & Suárez, S. (2011). Recent Advances in Signal Processing, Computational Geometry and Systems Theory. Retrieved from https://www.researchgate.net/publication/262212304_About_theory_in_software_development? On May, 5th 2020

[2] Pressman, R.S. Software Engineering A Practitioner's Approach. USA: The Mcgraw-Hill, 2010

[3] Partridge D. Artificial Intelligence and Software Engineering Understanding the promise of the future Routledge Taylor and Francis group, 1998 https://books.google.com.tr/books?id=ncBmAgAAQBAJ&printsec=frontcover#v=onepage&q&f=false

[4] F. Chollet, Deep Learning with Python. USA: Manning Publications, 2018

[5] Jörg, R. Klaus-Dieter, A. (2004). Retrieved from https://www.researchgate.net/publication/220633840_Artificial_Intelligence_and_Software_Engineering_Status_and_Future_Trends on April, 27th 2020

[6] Russell, S. Dewey, D. Tegmark, M. (2005). Research Priorities for Robust and Beneficial Artificial Intelligence, Copyright © 2015, Association for the Advancement of Artificial Intelligence. All rights reserved. ISSN 0738-4602 Retrieve from https://futureoflife.org/data/documents/research_priorities.pdf on April, 24th 2020

[7] Mostow J. (1985). What is AI? And What Does It Have to Do with Software Engineering? IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. SE-li, NO. 11, NOVEMBER 1985. Retrieve from https://s3.amazonaws.com/ieeecs.cdn.csdl.content/trans/ts/1985/11/01701944.pdf? on May, 2th 2020

[8] Retrieved from https://becominghuman.ai/amazons-sexist-ai-recruiting-tool-how-did-it-go-so-wrong-e3d14816d98e on May, 11th 2020

[9] Hany, H. A. Walid, Abdelmoez. & Mohamed S. H. (2012) Software Engineering Using Artificial Intelligence Techniques: Current State and Open Problems. ICCIT 2012 Retrieve from https://d1wqtxts1xzle7.cloudfront.net/30814527/p24-ammar.pdf?1362360177=&response-content-disposition=inline%3B+filename%3DSoftware_Engineering_Using_Artificial_In.pdf on May, 2nd 2020

[10] Mark H. (2012). The Role of Artificial Intelligence in Software Engineering. 2012 IEEE, RAISE Zurich, Switzerland. 978-1-4673-1753-5/12/$31.00 c? Retrieve from https://www.computer.org/csdl/pds/api/csdl/proceedings/download-article/ on May, 4th 2021

[11] Sunil, M. Thomas, K. & Jonathan, W. (2018) Artificial Intelligence and IT Professionals Published by the IEEE Computer Society 1520-9202/18/$33.00©2018 Retrieve from https://s3.amazonaws.com/ieeecs.cdn.csdl.content/mags/it/2018/05/mit2018050006.pdf? On May, 4th 2020

[12] K. Goertzel, “Legal Liability for Bad Software” Research Gate, USA. Accessed on April, 2nd 2021 https://www.researchgate.net/publication/309429972_KM_Goertzel_Legal_Liability_for_Bad_Software_CrossTalk_Vol_29_No_5_2016-0910

[13] Nees, M. (2016). Acceptance of Self-driving Cars: An Examination of Idealized versus Realistic Portrayals with a Self- driving Car Acceptance Scale. Proceedings of the Human Factors and Ergonomics Society Annual Meeting. 60. 1449-1453. 10.1177/1541931213601332.Retrieved from https://www.researchgate.net/publication/308179113_Acceptance_of_Self-driving_Cars_An_Examination_of_Idealized_versus_Realistic_Portrayals_with_a_Self-_driving_Car_Acceptance_Scale on June, 12th 2021

[14] National Museum of American History. Retrieved from https://americanhistory.si.edu/america-on-the-move/essays/automobile-safety on June 20th 2021

[15] T. Mustafa and A. Varol, "Review of the Internet of Things for Healthcare Monitoring," 2020 8th International Symposium on Digital Forensics and Security (ISDFS), 2020, pp. 1-6, doi: 10.1109/ISDFS49300.2020.9116305.

## AUTHORS

**Sabeer Saeed** is, currently, a Master's student in the department of Software Engineering at Firat University Elazig, Turkey. Before his coming to Firat University, He obtained International Advance (and Diploma) Diploma in computing from Informatics Academy. He also obtained B.Tech. in Management Information Technology (MIT) at Abubakar Tafawa Balewa University (ATBU), Bauchi, Bauchi Nigeria. Sabeer Saeed is a winner of National Information Technology Development Agency (NITDA)’s study scholarship to Firat University, Turkey where is studying Master’s in Software Engineering. He is a humanitarian par excellence, especially, on technology and gender diversity. The same humanitarian passion drove made him to establish an ICT Centre for School for Continuing Education (Women) at the Zamfara State, Nigeria during his one-year mandatory National Youth Service.