Mining the Powers of Open Source
A Case-Study on Usage of Open Source in Institutions

Dr. S. Mercy Shalinie
Professor, Department of Computer Sci. & Engg
Thiagarajar College of Engineering Madurai, India
shalinie@tce.edu

and

Ms. Nagarathna Ravi
PhD scholar, Department of Computer Sci. & Engg
Thiagarajar College of Engineering Madurai, India
rathnaravi2013@gmail.com

Abstract

Free and Open Source Software (FOSS) has steadily penetrated into the cyber world owing to the multifarious advantages it offers. Usage of FOSS can be found in various products such as internet browsers, web servers, e-commerce platforms etc. In this paper, we have given a report on how FOSS can be used in institutions effectively. In order to ascertain the effectiveness of FOSS in institutions, we have provided the arenas in which FOSS has been adopted in our institution, Thiagarajar College of Engineering (TCE) Madurai.

I. Introduction

Free and Open Source Software (FOSS) are products developed for users by users [1]. FOSS users form communities and develop the products without the involvement of the manufacturer. This provides multifarious advantages. From the view point of economy, FOSS reduces CAPEX and OPEX to great extent [2]. Commercial products are made to satisfy people at large and increase their profit. As FOSS users have direct access to the source code, they can feel free to experiment on the source code and custom craft it as per their needed. Hence, innovation is guaranteed with FOSS and also vendor lock in is completely evicted in FOSS. FOSS is easily adaptable to any new use cases. Since source code is available, bugs can be easily rectified as and when it rises. It also provides a great opportunity for developers to showcase their talent and gain experience of developing a software. There are numerous merits of FOSS. Owing to these merits, FOSS has been used and accepted widely by the cyber community.

FOSS finds usage in various applications starting from servers to mission critical hardware [3]. Various commonly used examples of FOSS are [3]- internet infrastructure technology such as Perl, MySQL, PHP, Apache; internet browser such as chrome, Firefox; server and desktop operating system such as Linux, Haiku, FreeBSD, NetBSD; desktop application software such as GIMP, Blender, OpenOffice; web applications such as WordPress, Mediawiki; python programming language; email client such as Thunderbird, Sendmail; DNS server software BIND; typesetting software TeX etc.

FOSS is also widely used in various academic institutions. For instance, virtual learning system like Moodle, learning management system like Sakai are used for betterment of students. Various institutions have also developed their own FOSS tools. For instance, Stanford University has developed FOSS tool named Stanford University Unstructured (SU^2) using open source C++ collection, to analyse complex multi-physics problems and solve constraint optimization, partial differential equation problems [5]. RWTH Aachen University has developed a speech recognition system open source tool [6]. Similarly, various academic institutions have adopted FOSS. We have also used FOSS to automate the activities in our institution Thiagarajar College of Engineering (TCE) Madurai. In this paper, we will give a detailed report on the usage of FOSS in our institution to show the benefits of FOSS.

The rest of this paper is organized as follows. Section II gives an overview of the objectives and activities of the open source group in TCE. Section III states the various research work undertaken by the members of the open source group. Section IV explains the projects done by open source members for use within the TCE campus. Section V gives a concluding note and future planned activities by the open source group.

II. Open Source Group at TCE

TCE has an open source community known as GNU/Linux User Group (GLUGOT). GLUGOT was started in December 2003 with an objective to promote FOSS and develop innovative spirit of cooperation among the students. GLUGOT also has more objectives. GLUGOT aims at developing research projects using FOSS. GLUGOT also strives to collaborate with National and International Universities and Industries to develop open source projects. GLUGOT has a mailing list named http://lists.tce.edu hosted on the TCE Mail server. GLUGOT also contributes to the community in the form of awareness

meetings, technical meetings and training program. In addition it organizes discussion groups, configuration and bug squashing sessions.

III. GLUGOT Research activities

GLUGOT group members are actively involved in various research activities using FOSS. The various research domains in which GLUGOT is active are network security, Natural Language Processing (NLP) and parallel algorithms.

GLUGOT SDN security research team utilizes various open source tools to research on the next generation network paradigms like Software Defined Network (SDN) and Information Centric Network (ICN). SDN research team has built a testbed using openflow protocol and open source controllers (RYU, POX) [7]. SDN research team has also come up with solutions for various security breaches using open source emulator Mininet and Mininet-WiFi [8][9]. ICN research team has come up with solutions for security breaches in ICN using open source simulator NDNSim [10][11][12].

In the field of NLP research is going on in various domains like development of tools to localize Tamil language, categorize the text and efficient document representation using FOSS. Semantics is a key process in NLP. Semantics help the computer to elicit the meaning of natural language of human. Negative words have a deep impact on a sentence. It is pretty hard for computer to elicit the meaning of a negation sentence. Research team has come up with various solutions to negation semantics using FOSS [13]. One more challenging task in NLP is to pick up relevant response to a query from a huge corpus of documents. Solutions to this issue is also completely got using FOSS by NLP researchers in TCE [14]. Yet another challenging task in NLP is reading comprehension. In this, the computer has to understand a given passage and answer the questions raised by the users. Research team in TCE has utilized a machine learning approach to impart intelligence to computer using an open source machine learning library named TensorFlow [15][16].

Research is going on in the field of parallel algorithms. We have GPU computing modules powered by FOSS to research on parallel scientific engineering problems. We have high performance cluster comprising of 1 master and 32 slave nodes. Open source Message Passing Interface (MPI) provides defined APIs. Researchers at TCE have used open source to design parallel algorithms for organizing a large corpus of web pages for easy retrieval [17][18].

Similarly various research works are going on in different domains using FOSS. Researchers at TCE have utilized FOSS to bring about solutions for the most necessary day to day problems in society like water security [19]. Thus through FOSS, we are also able to give back most useful solutions to the societal problems.

IV. GLUGOT Projects within TCE

GLUGOT has automated several processes within TCE using FOSS. We will describe few processes in this section.

A. TCENet

TCENet is a FOSS based ERP package developed to automate the day-to-day activities of TCE from student admission to alumni association. It has 30 modules such as online news, attendance, alumni, online assignment, placement, software forum, poll, maintenance, e-circular, feedback, internal assessment, articles, dues, profiles, nominal rolls, time table, search, hot downloads, right now messages, thought for the moment, birthday wishes, calendar, online assignment, achievements, status report and video lectures. Figure 1 shows the screenshot of TCENet.
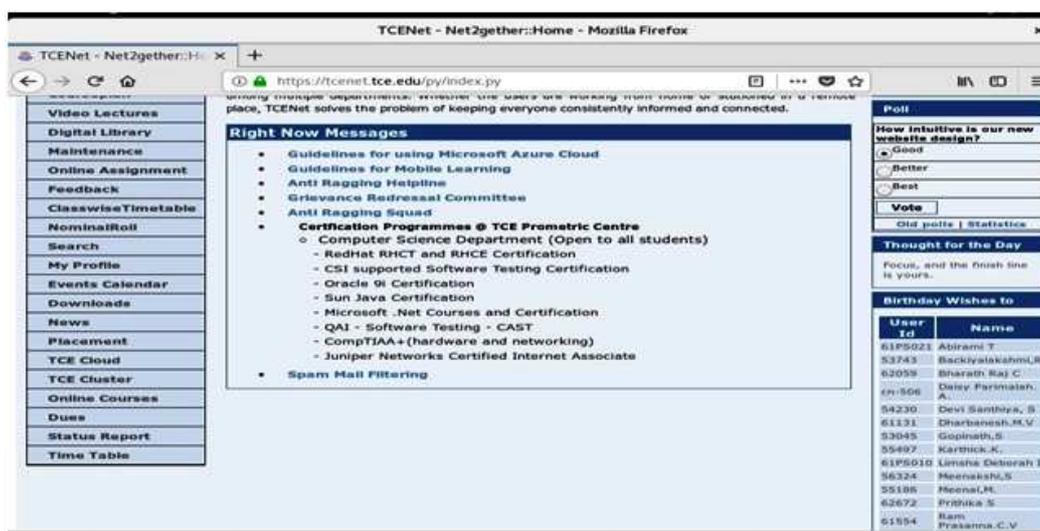

Fig. 1 TCENet page

The platform used is Debian 9. The web server used is uWSGI and nginx. AngularDart is used for client-side web app development framework. Pyramid framework is used for server-side web app development. PostgreSQL 9.6 is used for database. Template engine used is Jinja2. Python3.5 programming language is used. GitHub and bitbucket are used for version control system.

B. Single Sign-On Central Authentication System

The main policy is '1 user: 1 password' i.e. user access to all machines inside TCE campus in various laboratories using single user-id. All servers and machines inside the campus are integrated with centralized computing centre servers. A centralized storage box NAS box has been configured for users to store or upload files. Shell server has been configured for accessing local file storage outside TCE campus. OpenLDAP based Central Authentication System is configured. Samba based file server is used to authenticate with LDAP. A domain controller is configured in order to facilitate access from both GNU/Linux and other operating systems like Windows.

C. TCE Admission Automation

TCE admission automation is a web-based student admission process which includes students record creation, certificate verification, fee collection, course registration and report generation. This module also uses FOSS tools like Pyramid framework, bitbucket etc. Figure 2 shows a screenshot of the TCE admission portal.

Fig. 2 TCE Admission Automation

D. TCE Attendance Monitoring System

TCE attendance monitoring system is a combination of RFID and biometric technology. Various terminals capable of reading smart cards and finger prints are installed at every department building in TCE. These terminals are connected to the IP network infrastructure of TCE, thus giving instant access to the databases and servers of TCE. The time management and leave management are combined with attendance monitoring system. This module also purely runs using FOSS. Figure 3 shows a screenshot of the user portal of attendance monitoring system.

Fig. 3. TCE Attendance Monitoring System

E. TCE Website www.tce.edu

The main components of the website are home page with links to institute, departments, courses, admissions, activities, photo gallery, alumni, library, placement, intranet links, news, events, search and other information. All these are built and maintained with FOSS. Figure 4 shows a snapshot of the website.
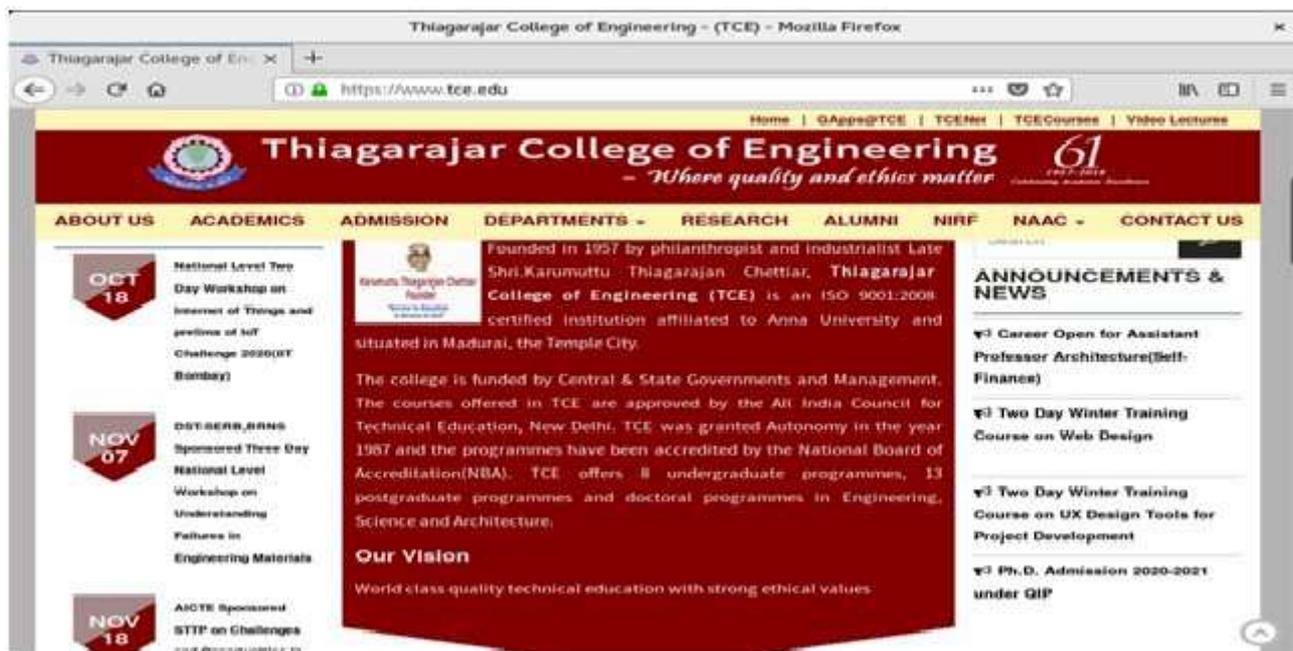

Fig. 4 *www.tce.edu*

F. TCE Firewall

We have a custom-made firewall built using iptables and bridge-utils. The commodity PC hardware runs no other service other than the firewall. NAT for internal networking is done using iptables in the TCE proxy server. Port forwarding and blocking is also included. TCE wide web cache using Squid is also used. To monitor the traffic load in each department in TCE, we have used Multi Router Traffic Grapher tool. This can effectively indicate unusual traffic surge. It is also possible to core down the source of traffic surge using the Multi Router Traffic Grapher tool. Figure 5 shows the snapshot of TCE traffic monitoring portal.
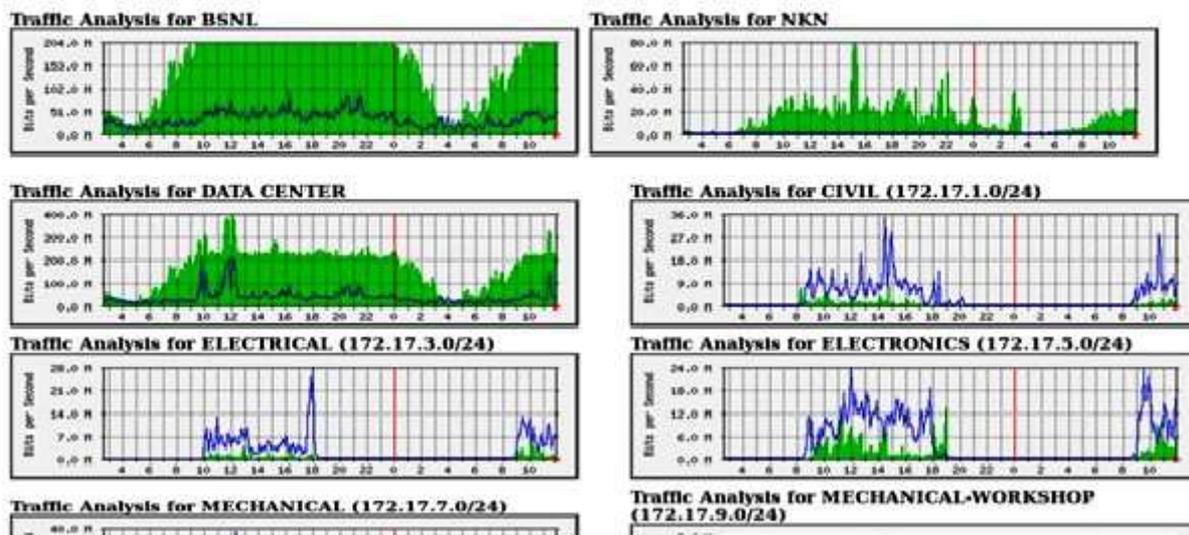

Fig. 5 TCE Traffic Analyzer

Similarly, almost all the activities are automated in TCE using FOSS. Examination related activities like pre-examination work, exam schedule preparation, exam attendance tracker, fees collection, revaluation applications, marks entry, result publishing etc. are automated using FOSS. TCE has its own datacenter which is completely automated using FOSS. TCE hosts a private cloud exclusively for usage of TCE faculty and students. TCE cloud was completely set using Openstack. Like this office activities are also automated using FOSS. Utilizing FOSS gave us various benefits. Prime advantage we had was cash freedom to use software and we were able to invest in hardware rather than software. We were able to customize each and every module based on our needs. Students were also given opportunities to code some modules which gave them a good experience. Using this experience various students were able to become entrepreneurs and had started their own company.

IV. Conclusion

In this paper we have briefed on the powers of open source by providing a case study of open source in institutions. We have provided the ways in which open source is used in our institution. Starting from research to automation of campus activities, our institution utilizes open source. This is a live example showcasing the powers of open source. We plan to automate remaining manual activities in our campus using FOSS for improved efficacy in future.

References

[1] Von Hippel, Eric. "Learning from open-source software. *"MIT Sloan management review* 42, no. 4 (2001): 82-86.

[2] Bonaccorsi, Andrea, and Cristina Rossi. "Why open source software can succeed. *"Research policy* 32, no. 7 (2003): 1243-1258.

[3] Heron, Michael, Vicki L. Hanson, and Ian Ricketts. "Open source and accessibility: advantages and limitations. *"Journal of interaction Science* 1, no. 1 (2013): 2.

[4] http://oss-watch.ac.uk/resources/softwareexamples

[5] Palacios, Francisco, Juan Alonso, Karthikeyan Duraisamy, Michael Colonno, Jason Hicken, Aniket Aranake, Alejandro Campos et al. "Stanford university unstructured (su 2): an open-source integrated computational environment for multi-physics simulation and design." In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, p. 287. 2013.

[6] Rybach, David, Christian Gollan, Georg Heigold, Björn Hoffmeister, Jonas Lööf, Ralf Schlüter, and Hermann Ney. "The RWTH Aachen University open source speech recognition system." In *Tenth Annual Conference of the International Speech Communication Association*. 2009.

[7] Ravi, Nagarathna, and S. Mercy Shalinie. "TeFENS: Testbed For Experimenting Next-Generation-Network Security." In *2018 IEEE 5G World Forum (5GWF)*, pp. 204-209. IEEE, 2018.

[8] Ravi, Nagarathna, and S. Mercy Shalinie. "SLAMHHA: A supervised learning approach to mitigate host location hijacking attack on SDN controllers." In *2017 Fourth International Conference on Signal Processing, Communication and Networking (ICSCN)*, pp. 1-7. IEEE, 2017.

[9] Ravi, Nagarathna, Rani, P. Vimala, and S. Mercy Shalinie. "Secure Deep Neural (SeDeN) Framework for 5G Wireless Networks." In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, 2019.

[10] Rani, P. Vimala, Nagarathna Ravi, S. Mercy Shalinie, P. Pariventhan, and K. Rajkumar. "Fuzzy Based Congestion-Aware Secure (FuCaS) Route Selection in ICN." In *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1-7. IEEE, 2018.

[11] Rani, P. Vimala, Nagarathna Ravi, S. Mercy Shalinie, P. Pariventhan, and K. Rajkumar. "Secure Framework for Information Centric Network over IoT Architecture." In *National Conference on Machine Learning and Artificial Intelligence*, 2018.

[12] Rani, P. Vimala, Nagarathna Ravi, S. Mercy Shalinie, and P. Pariventhan. "Detecting and Assuaging Against Interest Flooding Attack Using Statistical Hypothesis Testing in Next Generation ICN." In *2018 International Conference on Computer, Communication, and Signal Processing (ICCCSP)*, Chennai, pp. 1-5, IEEE, 2018.

[13] J. Felicia Lilian, K. Sundarakantham, Harine Rajashree, and S. Mercy Shalinie. "SSE: Semantic Sentence Embedding for Learning User Interactions." In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, 2019.

[14] J. Felicia Lilian, K. Sundarakantham, and S. Mercy Shalinie. "An Efficient Information Retrieval Mechanism From Relevant Document." In *1st International Symposium on Artificial Intelligence and Computer Vision*, 2018.

[15] K. Sundarakantham, J. Felicia Lilian, Harine Rajashree, and S. Mercy Shalinie. "ACA: Attention based Context-Aware Answer Selection System." In *International Conference on Machine Intelligence and Signal Processing*, 2018.
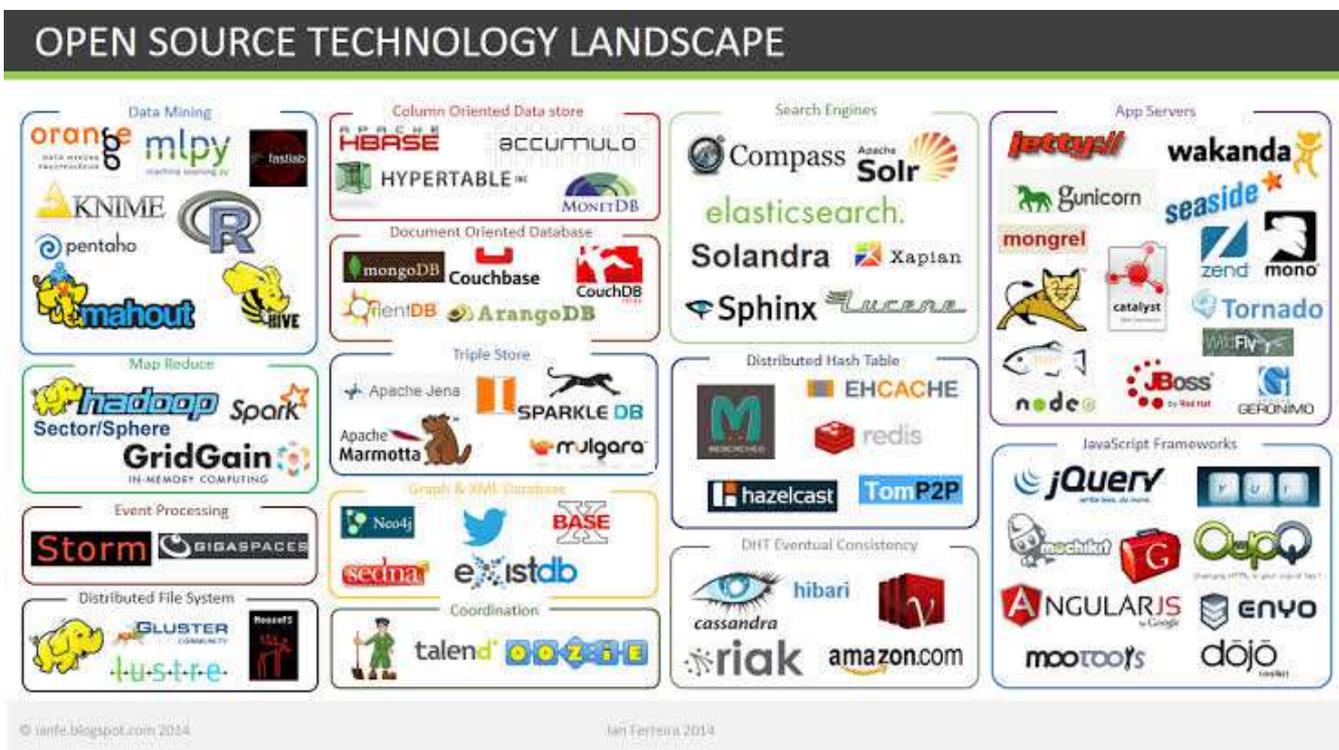
[16] J. Felicia Lilian, K. Sundarakantham, and S. Mercy Shalinie. "DLRCNeg: Deep Learning based Reading Comprehension by handling Negation." In *25th annual* **International Conference on Advanced Computing and Communications (ADCOM 2019),** 2019.

[17] Seshadri, Karthick, S. Mercy Shalinie, and Chidambaram Kollengode. "Design and evaluation of a parallel algorithm for inferring topic hierarchies." In *Information Processing & Management,* vol. 51, no. 5, pp. 662-676, 2015.

[18] Seshadri, Karthick, S. Mercy Shalinie, and Sidharth Manohar. "A distributed parallel algorithm for inferring hierarchical groups from large   scale text corpuses." In *Concurrency and Computation: Practice and Experience,* vol. 30, no. 11, e4404, 2018.

[19] R. H. Rajasekaran, S. Mercy Shalinie and J. Ramanathan. "A Novel Data-Driven DSSE Method to Achieve Water Sustainability for Farmers in Madurai District, India." In *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Bangalore, pp. 1-7, 2018.

S. Mercy Shalinie is working as Professor of the Department of Computer Science and Engineering, Thiagarajar College of Engineering, Madurai, Tamil Nadu, India. She has published over 200 research papers in referred journals and reputed conferences. She has received Scientist award in Engineering and Technology domain in Tamil Nadu, India. She was principal investigator for various sponsored research projects. Her area of research interest includes machine learning and security systems. She is Senior Member of IEEE.

Nagarathna Ravi is PhD scholar in Thiagarajar College of Engineering, Madurai, Tamil Nadu, India. She has published around 10 papers in journals and conferences. She was also session chair in IEEE 5G World Forum, Santa Clara, California, USA. Her research interest are machine learning and 5G network security which includes Software Defined Network and IoT. She is also interested in utilizing technology to bring about a smart environment.

Source: https://in.pinterest.com/