



Dr. Satwinder Singh

Associate Professor and Head,
Department of Computer Science and Technology,
Central University of Punjab, Ghudda, Punjab, India

📍 Room: 216, Academic Building, Central University of Punjab, Guddha, Punjab, India

☎ +91 93000 64064

✉ satwinder.singh@cup.edu.in, satwindercse@gmail.com

Area of Interest: *Software Engineering, Big Data, Machine Learning, and Natural Language Processing.*

WORK EXPERIENCE

Jan., 2020 – Present

Associate Professor

Central University Of Punjab, Bathinda, Punjab, India

Dec., 2015 – Jan., 2020

Assistant Professor

Central University Of Punjab, Bathinda, Punjab, India

August, 2006 – Dec., 2015

Assistant Professor

Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib, Punjab, India

July, 2004 – August, 2006

Lecturer

Rayat Institute of Engg. & Info. Tech., Railmajra, Nawanshahr, Punjab, India

EDUCATION

2014

Doctor of Philosophy

Guru Nanak Dev University, Amritsar, Punjab, India

2004

Master of Technology

Guru Nanak Dev University, Amritsar, Punjab, India

RESEARCH EXPERIENCE

Citations 403

h-index 11

i10-index 12

(Source: Google Scholar on June 13, 2021)

PROFESSIONAL RECOGNITION

Awards

Appreciation Award honoured by Central University of Punjab in the year 2017.

Best Teachers Award privileged by Indian Society for Technical Education, New Delhi in the year 2017.

Best Teachers Award esteemed by Indian Society for Technical Education, New Delhi in the year 2014.

RESEARCH GUIDANCE

Ph. D. Enrolled: 02
Guided: 02

M. Tech. M. Tech. Dissertation Guided: 56

PUBLICATIONS

- Journal**
1. Kanozia, R., Arya, R., Singh, S., Ganghariya, G., & Narula, S. (2021). A Study on Fake News Subject Matter, Presentation Elements, Tools of Detection, and Social Media Platforms in India. *Asian Journal for Public Opinion Research*, 9, 48-82.
 2. Sharma, S., & Singh, S. (2021). Texture-Based Automated Classification of Ransomware. *Journal of The Institution of Engineers (India): Series B*, 102(1), 131-142.
 3. Singh, R., & Singh, S. (2021). Text Similarity Measures in News Articles by Vector Space Model Using NLP. *Journal of The Institution of Engineers (India): Series B*, 102(2), 329-338.
 4. Itoo, F., & Singh, S. (2020). Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection. *International Journal of Information Technology*, 1-9.
 5. Kaur, R., Kaur, N., & Singh, S. (2019). Enhanced routing protocol for VANET. *International Journal of Innovative Technology and Exploring Engineering*, Volume 8, Year 2019, Pages 555-560
 6. Kaur, R., Kaur, N., & Singh, S. (2019). Analysis of Optimization Parameters of OLSR Protocol for VANETs. *International Journal of Innovative Technology and Exploring Engineering*, Volume 8, Year 2019, Pages 555-560
 7. Singh, S., & Singla, R. (2019). Object-Oriented Metrics for Defect Prediction. In *Software Engineering* (pp. 305-318). Springer, Singapore.
 8. Kumar, A., Singh, S., & Kaur, G. (2019). Fake news detection of Indian and United States election data using machine learning algorithm. *International Journal of Innovative Technology and Exploring Engineering*, Volume 8, Year 2019, Pages 1559-1563.
 9. Ali Reshi, J., & Singh, S. (2018). Investigating the role of code smells in preventive maintenance. *Journal of Information Technology Management*, 10(4), 41-63.
 10. Singh, S., & Kaur, S. (2018). A systematic literature review: Refactoring for disclosing code smells in object oriented software. *Ain Shams Engineering Journal*, 9(4), 2129-2151.
 11. Reshi, J. A., & Singh, S. (2018). Predicting software defects through SVM: an empirical approach. *arXiv preprint arXiv:1803.03220*.
 12. Singh, S., & Singla, R. (2018). Defect prediction model of static code features for cross-company and cross-project software. *International Journal of Information Technology*, 1-9.
 13. Singh, S., & Singla, R. (2017). Classification of defective modules using object-oriented metrics. *International Journal of Intelligent Systems Technologies and Applications*, 16(1), 1-13.
 14. Kumari, N., & Singh, S. (2017). Improving smell prediction: Developing an Improved Model with Supervised Learning Techniques. *Indian journal of science and technology*, 10(24).
 15. Meena S. & Singh, S (2017). DIFFERENT TECHNIQUE OF EXTRACTING THRESHOLD VALUE FOR SOURCE CODE METRICS. *IJIRMF. ISSN – 2455-0620 Volume - 3, Issue - 6, June - 2017*.
 16. Kaur, S., & Singh, S. (2017). Prediction Model to Investigate Influence of Code Smells on Metrics in Apache Tomcat. *International Journal of Advanced Research in Computer Science*, 8(5).
 17. Kaur, J., & Singh, S. (2016). Neural network based refactoring area identification in software system with object oriented metrics. *Indian Journal of Science and Technology*, 9(10), 1-8.
 18. Kaur, V. & Rani, P. & Singh, S. (2016). VAMPIRE ATTACKS: EXPLORATION & CONSEQUENCES. *International Journal of Scientific and Engineering Research*. 7. 1203-1207.
 19. Kaur, S., & Singh, S. (2015). Influence of anti-patterns on software maintenance: A review. *International Journal of Computer Applications*, 975, 8887.
 20. Kaur, H., & Singh, S. (2015). Analysis of CK metrics thresholds to predict faults using log transformation'. *Analysis*, Volume 4, Pages 45–62.
 21. Kumar, B., & Singh, S. (2015). Code clone detection and analysis using software metrics and neural network-a literature review. *Complexity*, 1(2), 3.
 22. Singh, S., & Kahlon, K. S. (2014). Object oriented software metrics threshold values at quantitative acceptable risk level. *CSI transactions on ICT*, 2(3), 191-205.

23. Kaur, R., & Singh, S. (2014). Clone detection in software source code using operational similarity of statements. *ACM SIGSOFT Software Engineering Notes*, 39(3), 1-5.
24. Singh, S., & Kaur, R. (2014). Clone detection in UML class models using class metrics. *ACM SIGSOFT Software Engineering Notes*, 39(3), 1-3.
25. Kapila, H., & Singh, S. (2014). Bayesian inference to predict smelly classes probability in open source software. *International Journal of Current Engineering and Technology*, 4(3), 1724-1728.
26. Kapila, H., & Singh, S. (2013). Analysis of CK metrics to predict software fault-proneness using bayesian inference. *International Journal of Computer Applications*, 74(2).
27. Singh, P., Singh, S., & Kaur, J. (2013). Tool for generating code metrics for C# source code using abstract syntax tree technique. *ACM SIGSOFT Software Engineering Notes*, 38(5), 1-6.
28. Kaur, S., Singh, S., & Kaur, H. (2013). A quantitative investigation of software metrics threshold values at acceptable risk level. *International Journal of Engineering Research and Technology*, 2(3), 1-7.
29. Singh, S., Mittal, P., & Kahlon, K. S. (2013). Empirical model for predicting high, medium and low severity faults using object oriented metrics in Mozilla Firefox. *International journal of computer applications in technology*, 47(2-3), 110-124.
30. Mittal, P., Singh, S., & Kahlon, K. S. (2013). Empirical model for fault prediction using object-oriented metrics in mozilla firefox. *Int. J. Comput. Technol. Res*, 1(6), 151-161.
31. Singh, S., & Kahlon, K. S. (2012). Effectiveness of refactoring metrics model to identify smelly and error prone classes in open source software. *ACM SIGSOFT Software Engineering Notes*, 37(2), 1-11.
32. Singh, S., & Kahlon, K. S. (2011). Effectiveness of encapsulation and object-oriented metrics to refactor code and identify error prone classes using bad smells. *ACM SIGSOFT Software Engineering Notes*, 36(5), 1-10.
33. Kaur, J., Singh, S., Kahlon, K. S., & Bassi, P. (2010). Neural network-a novel technique for software effort estimation. *International Journal of Computer Theory and Engineering*, 2(1), 17.
34. Singh, S., & Kahlon, K. S. (2010). Static Analysis To Model & Measure OO Paradigms. *International Journal of Computer Applications*, 975, 8887.
35. Kaur, A., Singh, S., Kahlon, K. S., & Sandhu, P. S. (2010). Empirical Analysis of CK & MOOD Metric Suit. *International Journal of Innovation, Management and Technology*, 1(5), 447.
36. Sharma, V., Singh, S., & Kahlon, K. S. (2009). Comparative Performance Study of Improved Heap Sort Algorithm on Different Hardware. *Journal of Computer Science*, 5(7), 476.
37. Kaur, A., Singh, S., & Kahlon, K. S. (2009). A metric framework for analysis of quality of object oriented design. *Int J Comput Inf Eng*, 3(12), 2875-2878.
38. Sharma, V., Sandhu, P. S., Singh, S., & Saini, B. (2008). Analysis of modified heap sort algorithm on different environment. *World Academy of Science, Engineering and Technology*, 42.
39. Sharma, V., Singh, S., & Kahlon, K. S. (2008). Performance study of improved Heap Sort algorithm and other sorting algorithms on different platforms. *IJCSNS*, 8(4), 101.
40. Kaur, J., Singh, S., & Kahlon, K. S. (2008). Comparative analysis of the software effort estimation models. *World Academy of Science, Engineering and Technology*, 46, 485-487.
41. Singh, S., Kahlon, K. S., Kumar, R., & Singh, G. (2005). Translator Design to Model Cpp Files. In *WEC* (5) (pp. 18-21).
42. Reshi, J. A., & Singh, S. Predicting Software Faults Using Software Metrics: A Review.
43. Reshi, J. A., & Singh, S. Object-Oriented Software Metrics And Machine Learning: An Empirical Study.
44. Bawa, A., & Singh, S. ANALYSIS OF SOFTWARE SYSTEM TO PREDICT ANTI-PATTERN WITH BAYESIAN INFERENCE.

- Conferences**
1. Paul, S., & Singh, S. (2020, November). Soil Moisture Prediction Using Machine Learning Techniques. In 2020 The 3rd International Conference on Computational Intelligence and Intelligent Systems (pp. 1-7).
 2. Singh, S., & Singla, R. (2016, March). Comparative performance of fault-prone prediction classes with k-means clustering and MLP. In Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (pp. 1-7).
 3. Mittal, A., & Singh, S. (2014). Survey on Various Security Attacks and the Mitigation Techniques for MANET. In International Conference on Communication, Computing and Systems (ICCCS) (Vol. 2, pp. 42-45).
 4. Sandhu, P. S., Singh, S., & Budhija, N. (2011, July). Prediction of level of severity of faults in software systems using density based clustering. In 2011 IEEE International Conference on Software and Computer Applications. IPCSIT (Vol. 9).
 5. Mittal, P., Singh, S., & Kahlon, K. S. (2011, July). Identification of error prone classes for fault prediction using object oriented metrics. In International Conference on Advances in Computing and Communications (pp. 58-68). Springer, Berlin, Heidelberg.
 6. Singh, S., Kahlon, K. S., & Sandhu, P. S. (2010, April). Re-engineering to analyze and measure object oriented paradigms. In 2010 2nd IEEE International Conference on Information Management and Engineering (pp. 472-478). IEEE.