Robert Dyer
Assistant Professor
University of Nebraska–Lincoln
School of Computing
103E Avery Hall
Lincoln, NE 68588 USA

curriculum vitae

Phone: (402) 472-5082
E-mail: rdyer@unl.edu
Web: `https://go.unl.edu/rdyer`

1 Education and Employment History

1.1 Education History

Iowa State University Ames, IA
Ph.D., Computer Science December 2013
Dissertation: Bringing Ultra-Large-Scale Software Repository Mining to the Masses with Boa
Adviser: Dr. Hridesh Rajan
Area of Study: Software Engineering

Iowa State University Ames, IA
M.S., Computer Science December 2008
Thesis: Supporting Dynamic Aspect-oriented Features
Adviser: Dr. Hridesh Rajan
Area of Study: Software Engineering

Iowa State University Ames, IA
B.S., Computer Science and Computer Engineering December 2006

1.2 Employment History

University of Nebraska–Lincoln (UNL) Lincoln, NE

- **Assistant Professor** August 2021–current
  School of Computing
- **Assistant Professor** August 2020–August 2021
  Department of Computer Science and Engineering

Iowa State University (ISU) Ames, IA

- **Affiliate Assistant Professor** January 2021–December 2023
  Department of Computer Science

Bowling Green State University (BGSU) Bowling Green, OH

- **Associate Professor** August 2020
  Department of Computer Science
- **Assistant Professor** August 2014–August 2020
  Department of Computer Science

Iowa State University (ISU) Ames, IA

- **Postdoctoral Researcher** January–June 2014
  Department of Computer Science

2 Research Accomplishments

2.1 Publication Record

2.1.1 Peer Reviewed Journal Publications in Print

[J9] F. Batole, R. Manke, **R. Dyer**, T. N. Nguyen, and H. Rajan, "Typestate-based fault localization of API usage violations in a deep learning program," *IEEE Transactions on Software Engineering*, p. to appear, 2026

[J8] S. W. Flint, J. Chauhan, N. Mansoor, **R. Dyer**, and B. Sharif, "An exploratory eye tracking study on how developers classify and debug Python code in different paradigms," *Empirical Software Engineering*, p. to appear, 2025

[J7] S. Perez-Rosero, **R. Dyer**, S. W. Flint, S. McIntosh, and W. Srisa-An, "WIASZZ: Work item aware SZZ," *Empirical Software Engineering*, vol. 30, no. 75, 2025. doi: 10.1007/s10664-025-10616-2

[J6] S. W. Flint, A. M. Keshk, **R. Dyer**, and H. Bagheri, "How do developers use type inference: An exploratory study in Kotlin," *Empirical Software Engineering*, vol. 30, no. 55, 2025. doi: 10.1007/s10664-024-10585-y

[J5] S. W. Flint, J. Chauhan, and **R. Dyer**, "Pitfalls and guidelines for using time-based Git data," *Empirical Software Engineering*, vol. 27, no. 7, 2022. doi: 10.1007/s10664-022-10200-y

[J4] M. Bagherzadeh, **R. Dyer**, R. D. Fernando, J. Sánchez, and H. Rajan, "Modular reasoning in the presence of event subtyping," *Transactions on Modularity and Composition I*, vol. 1, no. 1, pp. 167–223, 2016. doi: 10.1007/978-3-319-46969-0_5

[J3] **R. Dyer**, H. A. Nguyen, H. Rajan, and T. N. Nguyen, "Boa: Ultra-large-scale software repository and source-code mining," *ACM Transactions on Software Engineering and Methodology*, vol. 25, no. 1, pp. 7:1–7:34, 2015. doi: 10.1145/2803171

[J2] **R. Dyer**, H. Rajan, and Y. Cai, "Language features for software evolution and aspect-oriented interfaces: An exploratory study," *Transactions on Aspect-Oriented Software Development X*, vol. 1, no. 1, pp. 148–183, 2013. doi: 10.1007/978-3-642-36964-3_5

[J1] **R. Dyer** and H. Rajan, "Supporting dynamic aspect-oriented features," *ACM Transactions on Software Engineering and Methodology*, vol. 20, no. 2, 2010. doi: 10.1145/1824760.1824764

2.1.2 Books and Book Chapters

[B1] **R. Dyer**, H. A. Nguyen, H. Rajan, and T. N. Nguyen, "Boa: an enabling language and infrastructure for ultra-large scale MSR studies," in *The Art and Science of Analyzing Software Data*, C. Bird, T. Menzies, and T. Zimmermann, Eds. Waltham, MA: Morgan-Kaufmann, 2015, pp. 593–621

2.1.3 Conference Proceedings: Peer Reviewed

[C32] S. W. Flint, **R. Dyer**, and B. Sharif, "Do developers read type information? An eye-tracking study," in *Proceedings of the IEEE International Conference on Program Comprehension*, ser. ICPC'26, 2026, p. to appear

[C31] C. Moloney, **R. Dyer**, and E. Sherman, "Demonstrating ARG-V's generation of realistic Java benchmarks for SV-COMP," in *Proceedings of the International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, ser. TACAS'26, 2026, p. to appear

[C30] I. Abdelmadjid and **R. Dyer**, "pyMethods2Test: A dataset of Python tests mapped to focal methods," in *Proceedings of the International Conference on Mining Software Repositories*, ser. MSR'25, Ottawa, Canada, 2025. doi: 10.1109/MSR66628.2025.00123 p. accepted

[C29] F. Batole, D. OBrien, T. N. Nguyen, **R. Dyer**, and H. Rajan, “An LLM-based agent-oriented approach for automated code design issue localization,” in *Proceedings of the IEEE/ACM International Conference on Software Engineering Companion*, ser. ICSE, Ottawa, Canada, 2025. doi: 10.1109/ICSE55347.2025.00100 p. accepted

[C28] D. OBrien, **R. Dyer**, T. N. Nguyen, and H. Rajan, “Data-driven evidence-based syntactic sugar design,” in *Proceedings of the 46th IEEE/ACM International Conference on Software Engineering*, ser. ICSE, Lisbon, Portugal, 2024. doi: 10.1145/3597503.3639580 pp. 1–12

[C27] B. Sigurdson, S. W. Flint, and **R. Dyer**, “Boidae: Your personal mining platform,” in *Proceedings of the 46th IEEE/ACM International Conference on Software Engineering*, ser. ICSE, Lisbon, Portugal, 2024. doi: 10.1145/3639478.3640026 p. to appear

[C26] A. M. Keshk and **R. Dyer**, “Method chaining redux: An empirical study of method chaining in Java, Kotlin, and Python,” in *Proceedings of the 20th International Conference on Mining Software Repositories*, ser. MSR, Melbourne, Australia, 2023. doi: 10.1109/MSR59073.2023.00080 pp. 546–557

[C25] **R. Dyer** and J. Chauhan, “An exploratory study on the predominant programming paradigms in Python code,” in *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE. Singapore: Association for Computing Machinery, 2022. doi: 10.1145/3540250.3549158. ISBN 9781450394130 pp. 684–695

[C24] **R. Dyer** and S. W. Flint, “Performing large-scale mining studies: From start to finish (tutorial),” in *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE. Singapore: Association for Computing Machinery, 2022. doi: 10.1145/3540250.3569448. ISBN 9781450394130 p. 1822

[C23] S. W. Flint, J. Chauhan, and **R. Dyer**, “Escaping the time pit: Pitfalls and guidelines for using time-based Git data,” in *Proceedings of the 18th International Conference on Mining Software Repositories*, ser. MSR, Madrid, Spain, 2021. doi: 10.1109/MSR52588.2021.00022 pp. 85–96

[C22] C. S. Hung and **R. Dyer**, “Boa views: Easy modularization and sharing of MSR analyses,” in *Proceedings of the 17th International Conference on Mining Software Repositories*, ser. MSR. Seoul, Republic of Korea: Association for Computing Machinery, 2020. doi: 10.1145/3379597.3387480. ISBN 9781450375177 pp. 147–157

[C21] R. Brunner, **R. Dyer**, M. Paquin, and E. Sherman, “PAClab: A program analysis collaboratory,” in *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE. Virtual Event, USA: Association for Computing Machinery, 2020. doi: 10.1145/3368089.3417936. ISBN 9781450370431 pp. 1616–1620

[C20] E. Sherman and **R. Dyer**, “Software engineering collaboratories (SEClabs) and collaboratories as a service (CaaS),” in *Proceedings of the 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ser. ESEC/FSE. Lake Buena Vista, FL, USA: Association for Computing Machinery, 2018. doi: 10.1145/3236024.3264839. ISBN 9781450355735 pp. 760–764

[C19] J. Su, M. Arafat, and **R. Dyer**, “Using consensus to automatically infer post-conditions,” in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings*, ser. ICSE. Gothenburg, Sweden: Association for Computing Machinery, 2018. doi: 10.1145/3183440.3195096. ISBN 9781450356633 pp. 202–203

[C18] S. Shekarforoush, R. Green, and **R. Dyer**, “Classifying commit messages: A case study in resampling techniques,” in *International Joint Conference on Neural Networks*, ser. IJCNN, Anchorage, AK, USA, 2017. doi: 10.1109/IJCNN.2017.7965999 pp. 1273–1280

[C17] H. Rajan, T. N. Nguyen, G. T. Leavens, and **R. Dyer**, "Inferring behavioral specifications from large-scale repositories by leveraging collective intelligence," in *Proceedings of the 37th International Conference on Software Engineering*, ser. ICSE, vol. 2, Firenze, Italy, 2015. doi: 10.1109/ICSE.2015.339 pp. 579–582

[C16] **R. Dyer**, H. Rajan, T. N. Nguyen, and H. A. Nguyen, "Demonstrating programming language feature mining using Boa," in *Companion Proceedings of the 2015 ACM SIGPLAN International Conference on Systems, Programming, Languages and Applications: Software for Humanity*, ser. SPLASH. Pittsburgh, PA, USA: Association for Computing Machinery, 2015. doi: 10.1145/2814189.2814192. ISBN 9781450337229 pp. 13–14

[C15] H. A. Nguyen, **R. Dyer**, T. N. Nguyen, and H. Rajan, "Consensus-based mining of API preconditions in big code," in *Companion Proceedings of the 2015 ACM SIGPLAN International Conference on Systems, Programming, Languages and Applications: Software for Humanity*, ser. SPLASH. Pittsburgh, PA, USA: Association for Computing Machinery, 2015. doi: 10.1145/2814189.2816271. ISBN 9781450337229 pp. 5–6

[C14] M. Bagherzadeh, **R. Dyer**, R. D. Fernando, J. Sánchez, and H. Rajan, "Modular reasoning in the presence of event subtyping," in *Proceedings of the 14th International Conference on Modularity*, ser. MODULARITY. Fort Collins, CO, USA: Association for Computing Machinery, 2015. doi: 10.1145/2724525.2724569. ISBN 9781450332491 pp. 117–132

[C13] H. A. Nguyen, **R. Dyer**, T. N. Nguyen, and H. Rajan, "Mining preconditions of APIs in large-scale code corpus," in *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, ser. FSE. Hong Kong, China: Association for Computing Machinery, 2014. doi: 10.1145/2635868.2635924. ISBN 9781450330565 pp. 166–177

[C12] **R. Dyer**, H. Rajan, H. A. Nguyen, and T. N. Nguyen, "Mining billions of AST nodes to study actual and potential usage of Java language features," in *Proceedings of the 36th International Conference on Software Engineering*, ser. ICSE, Hyderabad, India, 2014. doi: 10.1145/2568225.2568295 pp. 779–790

[C11] **R. Dyer**, H. A. Nguyen, H. Rajan, and T. N. Nguyen, "Boa: A language and infrastructure for analyzing ultra-large-scale software repositories," in *Proceedings of the 35th International Conference on Software Engineering*, ser. ICSE, San Francisco, CA, 2013. doi: 10.1109/ICSE.2013.6606588 pp. 422–431

[C10] **R. Dyer**, "Task fusion: Improving utilization of multi-user clusters," in *Proceedings of the companion publication for conference on Systems, Programming, Languages, & Applications: Software for Humanity*, ser. SPLASH SRC, Indianapolis, IN, 2013. doi: 10.1145/2508075.2514878 pp. 117–118

[C9] **R. Dyer**, H. Rajan, and T. N. Nguyen, "Declarative visitors to ease fine-grained source code mining with full history on billions of AST nodes," in *Proceedings of the 12th International Conference on Generative Programming: Concepts & Experiences*, ser. GPCE, Indianapolis, IN, 2013. doi: 10.1145/2637365.2517226 pp. 23–32

[C8] **R. Dyer**, H. A. Nguyen, H. Rajan, and T. N. Nguyen, "Mining source code repositories with Boa," in *Proceedings of the 2013 Companion Publication for Conference on Systems, Programming, & Applications: Software for Humanity*, ser. SPLASH. Indianapolis, IN, USA: Association for Computing Machinery, 2013. doi: 10.1145/2508075.2514570. ISBN 9781450319959 pp. 13–14

[C7] **R. Dyer**, H. A. Nguyen, H. Rajan, and T. N. Nguyen, "Boa: Analyzing ultra-large-scale code corpus," in *Proceedings of the 3rd Annual Conference on Systems, Programming, and Applications: Software for Humanity*, ser. SPLASH. Tucson, AZ, USA: Association for Computing Machinery, 2012. doi: 10.1145/2384716.2384752. ISBN 9781450315630 pp. 87–88

[C6] **R. Dyer**, H. Rajan, and Y. Cai, “An exploratory study of the design impact of language features for aspect-oriented interfaces,” in *Proceedings of the 11th Annual International Conference on Aspect-Oriented Software Development*, ser. AOSD. Potsdam, Germany: Association for Computing Machinery, 2012. doi: 10.1145/2162049.2162067. ISBN 9781450310925 pp. 143–154

[C5] **R. Dyer**, H. A. Nguyen, H. Rajan, and T. N. Nguyen, “Analyzing ultra-large-scale code corpus with Boa,” in *Proceedings of the 3rd Annual Conference on Systems, Programming, and Applications: Software for Humanity*, ser. SPLASH. Tucson, AZ, USA: Association for Computing Machinery, 2012. doi: 10.1145/2384716.2384729. ISBN 9781450315630 pp. 25–26

[C4] H. Rajan, G. T. Leavens, **R. Dyer**, and M. Bagherzadeh, “Modularizing crosscutting concerns with Ptolemy,” in *Proceedings of the Tenth International Conference on Aspect-Oriented Software Development Companion*, ser. AOSD. Porto de Galinhas, Brazil: Association for Computing Machinery, 2011. doi: 10.1145/1960314.1960332. ISBN 9781450306065 pp. 61–62

[C3] H. Rajan, S. Mooney, G. T. Leavens, **R. Dyer**, R. D. Fernando, M. A. D. Darab, and B. Welter, “Modularizing crosscutting concerns with Ptolemy,” in *Proceedings of the ACM International Conference Companion on Object Oriented Programming Systems Languages and Applications Companion*, ser. OOPSLA. Portland, OR, USA: Association for Computing Machinery, 2011. doi: 10.1145/2048147.2048163. ISBN 9781450309424 pp. 31–32

[C2] **R. Dyer** and H. Rajan, “Nu: A dynamic aspect-oriented intermediate language model and virtual machine for flexible runtime adaptation,” in *Proceedings of the 7th International Conference on Aspect-Oriented Software Development*, ser. AOSD. Brussels, Belgium: Association for Computing Machinery, 2008. doi: 10.1145/1353482.1353505. ISBN 9781605580449 pp. 191–202

[C1] **R. Dyer**, H. Narayanappa, and H. Rajan, “Nu: Preserving design modularity in object code,” in *Proceedings of ACM SIGSOFT International Symposium on Foundations of Software Engineering*, ser. FSE. Portland, OR, USA: Association for Computing Machinery, 2006. doi: 10.1145/1218776.1218802 pp. 1–2

2.1.4 Conference Presentations

1. “pyMethods2Test: A Dataset of Python Tests Mapped to Focal Methods,” Conference on Mining Software Repositories, Idriss Abdelmadjid and Robert Dyer, April 2025, Canada.
2. “An exploratory study on the predominant programming paradigms in Python code,” 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, Robert Dyer and Jigyasa Chauhan, November 2023, Singapore.

2.1.5 Invited Talks/Keynote Speeches

1. “Leveraging Real-world Code to Learn About Programmer Behavior,” UNL, SOFT 160 (2021)
2. “Undergraduate Research,”, Iowa State University, CS192 (2021)
3. “Big Code Mining in Boa,” SDE lab, University of Macedonia, Greece (2021)
4. “Software Engineering Collaboratories as a Service,” University of Nebraska (2019)
5. “Scalable Tools,” Midwest Big Data Summer School (2019)
6. “Mining Thousands of Software Repositories,” Midwest Big Data Summer School (2019)
7. “An Investigation of Routine Repetitiveness in Open-Source Projects: A Partial Reproduction of "A large-scale study on repetitiveness, containment, and composability of routines in open-source projects",” ICSE ROSE Festival (2019)
8. “Introduction to Scalable Tools for Big Data,” Midwest Big Data Summer School (2016)
9. “Mining Programming Feature Usage at a Large Scale,” MSR Asia Summit (2015)

2.1.6 Other Publications

1. H. A. Nguyen, T. N. Nguyen, H. Rajan, and **R. Dyer**, "Towards combining usage mining and implementation analysis to infer API preconditions," in *Proceedings of the 1st ACM SIGSOFT International Workshop on Automated Specification Inference*, ser. WASPI. Lake Buena Vista, FL, USA: Association for Computing Machinery, 2018. doi: 10.1145/3278177.3278182. ISBN 9781450360579 pp. 15–16
2. A. Shakiba, R. Green, and **R. Dyer**, "FourD: Do developers discuss design? revisited," in *Proceedings of the 2nd International Workshop on Software Analytics*, ser. SWAN. Seattle, WA, USA: Association for Computing Machinery, 2016. doi: 10.1145/2989238.2989244. ISBN 9781450343954 pp. 43–46
3. R. D. Fernando, **R. Dyer**, and H. Rajan, "Event type polymorphism," in *Proceedings of the 11th Workshop on Foundations of Aspect-Oriented Languages*, ser. FOAL. Potsdam, Germany: Association for Computing Machinery, 2012. doi: 10.1145/2162010.2162020. ISBN 9781450310994 pp. 33–38
4. M. Bagherzadeh, G. T. Leavens, and **R. Dyer**, "Applying translucid contracts for modular reasoning about aspect and object oriented events," in *Proceedings of the 10th International Workshop on Foundations of Aspect-Oriented Languages*, ser. FOAL. Porto de Galinhas, Brazil: Association for Computing Machinery, 2011. doi: 10.1145/1960510.1960517. ISBN 9781450306447 pp. 31–35
5. **R. Dyer**, M. Bagherzadeh, H. Rajan, and Y. Cai, "A preliminary study of quantified, typed events," in *Empirical Evaluation of Software Composition Techniques, A workshop affiliated with AOSD*, ser. ESCOT, Rennes and Saint-Malo, France, 2010
6. W. Baelen, Y. Cai, **R. Dyer**, and H. Rajan, "Feature volatility assessment," in *14th International Conference on Software Product Lines, Volume 2*, ser. SPLC. Jeju Island, South Korea: Lancaster University, 2010, pp. 33–34
7. **R. Dyer** and H. Rajan, "A decision tree-based approach to dynamic pointcut evaluation," in *Proceedings of the 2nd Workshop on Virtual Machines and Intermediate Languages for Emerging Modularization Mechanisms*, ser. VMIL. Nashville, TN, USA: Association for Computing Machinery, 2008. doi: 10.1145/1507504.1507508. ISBN 9781605583846
8. H. Rajan, **R. Dyer**, Y. Hanna, and H. Narayanappa, "Preserving separation of concerns through compilation," in *Software Engineering Properties of Languages and Aspect Technologies, A workshop affiliated with AOSD*, ser. SPLAT, Bonn, Germany, 2006

2.2 Grantsmanship Record

2.2.1 Externally Funded Research Grants

[G1] NSF, 1 Jul 2024, *Collaborative Research: CIRC: Dev: Adaptable Realistic Benchmark Generator for Verification (ARG-V)*, Total: $732,656, UNL: $246,883 (apportionment: 100%), PI (UNL): Robert Dyer, PI (Boise State): Elena Sherman.

[G2] NSF, *CRI: CI-P: Towards a Program Analysis Collaboratory*, Total: $100,000, BGSU: $46,546 (apportionment: 100%), 08/2018–07/2020, PI (BGSU): Robert Dyer, PI (Boise State): Elena Sherman, NSF 1823294,

[G3] NSF, *SHF:Large:Collaborative Research: Inferring Software Specifications from Open Source Repositories by Leveraging Data and Collective Community Expertise*, Total: $1,604,479, BGSU: $214,843 (apportionment: 100%), 07/2015–06/2019, PI (BGSU): Robert Dyer, PI (Iowa State): Hridesh Rajan, PI (UCF): Gary Leavens, PI (Penn State): Vasant Honavar, NSF 1518776,

[G4] NSF, *CI-EN: Boa: Enhancing Infrastructure for Studying Software and its Evolution at a Large Scale*, Total: $1,559,806, BGSU: $132,889 (apportionment: 100%), 06/2015–05/2019, PI (BGSU): Robert Dyer, PI (Iowa State): Hridesh Rajan, NSF 1512947,

2.2.2 National and International Research Awards and Recognition

1. 2021 Best Paper Award: MSR'21
2. 2015 Best Paper Award: Modularity'15

2.2.3 Regional and Local Research Awards and Recognition

1. 2016 BGSU Outstanding Early Career Award
2. 2009 Dr Robert Stewart Early Research Award
3. 2007 CRA Outstanding Undergraduate Award Honorable Mention

2.3 Other Research Accomplishments

2.3.1 Infrastructure Available to Community

1. ARG-V–Adaptable Realistic Benchmark Generator for Verification, `https://arg-v.dev/`
2. Boa, over 1,400 registered users from 36 countries, service and source code made publicly available, `https://boa.cs.iastate.edu/`
3. Abdelmadjid, I., & Dyer, R. (2024). pyMethods2Test: A Dataset of Python Tests Mapped to Focal Methods (1.0.0) [Data set]. Zenodo. `https://doi.org/10.5281/zenodo.14264519`
4. Perez-Rosero, S., Dyer, R., Srisa-an, W., & McIntosh, S. (2024). A Dataset of Work Items (1.0.2) [Data set]. Zenodo. `https://doi.org/10.5281/zenodo.14279892`
5. PAClab–a Program Analysis ColLABoratory, service currently offline after moving from BGSU, but all source code is publicly available, `https://paclab.dev/`

3 Teaching Accomplishments (other than classroom instruction)

3.1 PhD Students

3.1.1 PhD Students Currently in Progress

1. Fatemeh Raei Dehaghi — January 2025–current
   - Computer Engineering, UNL
   - Dissertation: TBD
   - Expected graduation: May 2027

3.2 MS Students

3.2.1 MS Students (thesis option)

1. Jigyasa Chauhan — January 2021–December 2022
   - Computer Science, UNL
   - 100% funding
   - Graduated: December 2022
2. Rebecca Brunner — August 2019–August 2020
   - Computer Science, BGSU

  - 100% funding
  - Graduated: August 2020
3. Che Shian Hung August 2018–August 2019
   - Computer Science, BGSU
   - 100% funding
   - Graduated: August 2019
4. Mohd Arafat August 2016–August 2018
   - Computer Science, BGSU
   - 100% funding
   - Graduated: August 2018
5. Shubhendra Shrimal August 2014–August 2016
   - Computer Science, BGSU
   - 100% funding
   - Graduated: August 2016

3.2.2 Total Graduate Student Independent Research Projects Supervised

At Bowling Green State University, I supervised 11 MS student projects (non-thesis).

3.3 Undergraduate Students

3.3.1 Undergraduate Students Supervised in Independent Research Study

1. Charles Moloney August 2024–May 2025
   - Computer Science, UNL
   - Raikes School, Research Studio
2. A'laa Srour August 2024–December 2024
   - Visiting Egyptian student
   - 8 hours per week
3. Harry Do June 2023–December 2023
   - Software Engineering, UNL
   - 10 hours per week
   - 100% funding
4. Dylan Kramer August 2022–May 2023
   - Software Engineering, UNL
   - 10 hours per week
   - Funded by UNL First Year Research Experience (FYRE)
5. Kevin Tran August 2022–May 2023
   - Computer Science, UNL
   - 10 hours per week
   - Funded by UNL First Year Research Experience (FYRE)
6. Alisson Ntwali August 2022–May 2023
   - Integrated Science, UNL
   - 12 hours per week
   - 100% funding
7. Ali Keshk August 2021–July 2022

- Computer Science, UNL
- 10 hours per week academic year, 40 hours per week summer
- Academic year funded by UNL First Year Research Experience (FYRE)
- 100% funding during summer

8. Kareem Keshk January 2022–July 2022
   - Computer Science, UNL
   - 10 hours per week academic year, 40 hours per week summer
   - 100% funding
9. Aime Nishimwe January 2022–May 2022
   - Integrated Science, UNL
   - 20 hours per week
   - 100% funding
10. Parul Aggarwal May 2021–August 2021
    - Computer Science, UNL
    - Funded by UNL UCARE Award

3.4 Graduate Student Committee Membership

3.4.1 Doctoral Dissertation Committee

1. Mohammad Jalili Torkamani January 2026–current
   - Computer Science, UNL
   - Expected graduation: May 2028
2. Tony Arslan September 2023–current
   - Computer Science, UNL
   - Expected graduation: December 2026
3. Nedasadat Taheri March 2023–current
   - Engineering (Computer Engineering specialization), UNL
   - Expected graduation: December 2026
4. Kang-il Park August 2022–current
   - Computer Science, UNL
   - Expected graduation: December 2025
5. Lulah Alnaji August 2016–August 2018
   - Mathematics, BGSU
   - Graduated: August 2018

3.4.2 Masters Thesis Committee

1. Zachary Kozak September 2025–current
   - Computer Science, UNL
   - Expected graduation: May 2026
2. Mohammad Jalili Torkamani August 2025–December 2025
   - Computer Science, UNL
   - Graduated: December 2025
3. Jun Sun January 2021–August 2021
   - Computer Science, UNL

- Graduated: August 2021

4. Asanga Mudiyanselage January 2019–August 2020
   - Computer Science, BGSU
   - Graduated: August 2020
5. Blake Grills August 2019–May 2020
   - Computer Science, BGSU
   - Graduated: May 2020
6. Justin Kleinknecht August 2019–May 2020
   - Computer Science, BGSU
   - Graduated: May 2020
7. Jeremy Storer August 2015–May 2016
   - Computer Science, BGSU
   - Graduated: May 2016

4 SERVICE ACCOMPLISHMENTS

4.1 Professional Service

4.1.1 Journal Paper Reviewer

1. IEEE Transactions on Software Engineering (TSE) (2015–2020, 2022–2025)
2. Empirical Software Engineering (EMSE) (2014, 2016–2025)
3. ACM Transactions on Software Engineering and Methodology (TOSEM) (2024–2025)
4. Journal of Systems and Software (JSS) (2016–2018, 2021–2025)
5. ACM Transactions on Computing Education (TOCE) (2025)
6. Automated Software Engineering (ASE) (2024)
7. IEEE Software (2020)
8. Science of Computer Programming (SCP) (2020)
9. Software: Practice and Experience (SPE) (2014)
10. Transactions on Aspect-Oriented Software Development (TAOSD) (2010)

4.1.2 Conference Paper Reviewer

1. International Conference on Software Engineering (ICSE) (2026–2027)
2. Mining Software Repositories (MSR) (2017–2019, 2021–2026)
3. IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER) (2022–2026)
4. International Conference on Program Comprehension (ICPC) (2022–2024, 2026)
5. International Conference on Software Engineering (ICSE) Software Engineering in Society (SEIS) (2024–2026)
6. International Conference on Program Comprehension (ICPC) Tool Demonstrations (2025–2026)
7. International Conference on the Foundations of Software Engineering (FSE) (2025)
8. International Conference on Automated Software Engineering (ASE) (2023–2025)
9. International Conference on Software Maintenance and Evolution (ICSME) (2022–2025)
10. Innovations in Software Engineering Conference (ISEC) (2022–2023)
11. International Conference on Software Engineering (ICSE) Posters (2022)
12. International Conference on Software Maintenance and Evolution (ICSME) New Ideas and Emerging Results Track (2021)

13. External Review Committee member: ACM SIGPLAN conference on Systems, Programming, Languages, and Applications: Software for Humanity (SPLASH) OOPSLA (2020–2021)
14. Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE) (2020)
15. International Conference on Software Analysis, Evolution and Reengineering (SANER), Late Breaking Ideas track (2020)
16. Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE) Artifact Evaluation (2019)
17. International Conference on Software Maintenance and Evolution (ICSME) Tool Demonstrations (2019)
18. Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE) Formal Demonstrations (2018)
19. Automated Software Engineering (ASE) Tool Demonstrations (2016, 2018)
20. International Conference on Modularity (2015)
21. Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) Artifacts (2013–2015)
22. Student Research Competition (SRC), at Modularity (2014)

4.1.3 Leadership Positions in International and National Organizations

1. Organizing Committee: Co-Chair of New Faculty Symposium at Foundations of Software Engineering (FSE) (2026)
2. Organizing Committee: Co-chair of Registered Reports at International Conference on Software Maintenance and Evolution (ICSME) (2026)
3. Elected to ACM SIGSOFT Executive Committee, **Secretary-Treasurer** (2018–2021)
4. **Chair**: ACM SIGPLAN conference on Systems, Programming, Languages, and Applications: Software for Humanity (SPLASH) Doctoral Symposium (2021)
5. Organizing Committee: Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE) (2020)
6. Organizing Committee: ACM SIGPLAN conference on Systems, Programming, Languages, and Applications: Software for Humanity (SPLASH) (2020)
7. ACM SIGSOFT Webinar Coordinator (2015–2019)
8. Organizing Committee: Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE) (2018)
9. Organizing Committee: Workshop on Automated Specification Inference (WASPI), at ESEC/FSE (2018)
10. **Co-chair**: Mining Software Repositories (MSR) Mining Challenge (2016)
11. Organizing Committee: International Conference on Software Engineering (ICSE) (2016)
12. Organizing Committee: Workshop on Virtual Machines and Intermediate Languages (VMIL) (2009–2011)
13. Organizing Committee: Workshop on Virtual Machines and Intermediate Languages for emerging modularization mechanisms (VMIL) (2008)

4.1.4 Memberships in Professional Organizations

1. Association for Computing Machinery (ACM) (2006–current)
2. ACM Special Interest Group on Software Engineering (ACM SIGSOFT) (2006–current)
3. ACM Special Interest Group on Programming Languages (2006–current)

4.1.5 Research Review Panels

1. Israel Science Foundation reviewer (2021)
2. U.S. National Science Foundation (NSF) (2017, 2019, 2020)

4.2 University Service

4.2.1 Membership Positions on University Wide Committees

1. BGSU, Information Technology Committee (2018–2020)
2. BGSU, Graduate Awards Committee (2017–2019)
3. BGSU, ITS grants routing FMS application testing (2015, 2019)

4.3 College Service

4.3.1 Membership Positions on College Wide Committees

1. UNL, College Curriculum and Academic Standards Committee (CCASC) (2022–current)

4.4 Unit Service

4.4.1 Leadership Positions on Unit Committees

1. UNL, Colloquium Committee co-chair (2024–2025)
2. UNL, Undergraduate Curriculum Committee chair (Spring 2024)
3. BGSU, Acting Chair Search Committee chair (2017–2018)

4.4.2 Membership Positions on Unit Committees

1. UNL, SE Faculty Search Committee (2025–2026)
2. UNL, Broadening Participation in Computing Committee (2024–current)
3. UNL, Colloquium Committee (2024–current)
4. UNL, Graduate Committee, Admissions Subcommittee (2023–current)
5. UNL, Undergraduate Curriculum Committee (2022–current)
6. UNL, Graduate Committee, CS Subcommittee (2021–2022)
7. UNL, Awards Committee (2020–2021)
8. UNL, Graduate Committee, Graduate Recruitment Subcommittee (2020–2022)
9. BGSU, Executive Committee (2019–2020)
10. BGSU, Continuous Improvement Committee (2019–2020)
11. BGSU, Graduate Committee (2015–2019)
12. BGSU, Computer Science Advisory Board (CSAB) (2014–2020)
13. BGSU, ACM Faculty Advisor (2014–2019)
14. BGSU, Faculty Search Committee (2016–2017)
15. BGSU, Course Evaluations ad-hoc committee (2018)
16. BGSU, B.S. in Software Engineering degree ad-hoc committee (2015–2016)
17. BGSU, Preview Day recruitment (2014–2017)
18. BGSU, Commencement (2019, 2017, 2015, 2014)

5 Other Accomplishments

5.1 Professional Development

1. UNL, EVC, How to Write a Compelling Faculty Evaluation Narrative (Fall 2025)
2. UNL, Academic Technologies Learning Series, Polling App (Fall 2025)
3. UNL, CTT, Teaching with AI Skillshare, Learning Community (Spring 2025)
4. UNL, COE, Suicide Prevention Training (Spring 2025)
5. UNL, CTT, Designing Courses in the Age of AI workshop (Spring 2025)
6. UNL, CTT, CIMER Mentorship Training (Fall 2024)
7. UNL, CTT, Small Things, Big Impact: Setting the Stage for the Semester (Spring 2024)
8. UNL, CTT, Supporting Diverse Student Populations (Spring 2024)
9. UNL, COE, DEI Lunch and Learn (Fall 2023)
10. UNL, ECEC, Peer Observations of Classroom Activities (Fall 2022)
11. BGSU, CFE, Inclusive Pedagogy Workshop (Spring 2020)
12. BGSU, CFE, Guide to Teaching in the Active Learning Classroom Book Club (Spring 2019)
13. BGSU, OTN, Open Textbook Network Workshop (Fall 2017)
14. BGSU, CFE, Connecting Learning Expectations with Assessment (Fall 2016)
15. BGSU, Active Learning using Digital Technologies in the Classroom (Fall 2015–Spring 2016)
16. BGSU, CFE, Increase Student Engagement Using the Inverted Classroom (Fall 2015)
17. BGSU, Prism of Possibilities Conference (Fall 2015)
18. BGSU, CFE, Active Learning Classroom (Fall 2015)
19. BGSU, A&S Diversity Committee Faculty Mentoring Program (Fall 2015)
20. BGSU, New Faculty Learning Community (Fall 2014–Spring 2015)
21. BGSU, MyITLab Introduction (Fall 2014)
22. BGSU, Canvas 101, Canvas Basics (Fall 2014)
23. BGSU, Canvas 202, Online Course Design (Fall 2014)