Intelligent Software & Human Analytics (ISHA) Research Lab
Dr. Sridhar Chimalakonda

September 22, 2019

What will you learn through this exercise? Or Why bother?

- Rigorous problem solving under uncertainty, whether you are in computing or any disciplines!
- An engineering and research approach to solving challenging problems
- If you think you should do the way MIT/CMU does or even better, this is an attempt towards that!

Some Projects

Software Engineering

1. **Software Engineering Lab Projects** - As you know, most of the tools developed as part of software engineering lab are extensions of state-of-the-art tools and your work during the lab is only a first attempt. Some of them can be extended for BTP, but should be done in consultation with me. For example, tools such as Semantic Code Clones, Visualizing Software Architecture, Tools to support APIs, Code Smells, Extending Github/IDEs.
2. **Mining Software Repositories** - A project in this area typically has the following tasks:
    a. *Datasets and Empirical Studies* - Analyze software artifacts such as [code/bugs/user reviews/…] in data sources such as Github, StackOverflow, CodeChef to assess specific *qualities through metrics*. We are in the process of doing an MoU with *CodeChef*, which consists of millions of code submissions, which could be the dataset for some projects. For example, Are code smells in Github projects done across languages similar? Study the evolution of a particular software such as Google Chrome over multiple versions based on certain metrics. Can you figure out how many apps on google play store are really secure? What difficulties developers face when using APIs/frameworks and so on?
    b. *Design and development* of frameworks/platforms/tools/plugins to address the concerns identified in Step a, either from existing literature or from your own study. Tools include developer productivity tools, API improvement tools, code analysis tools, integration of Github and other data sources, tools to support visualization of software artifacts, their evolution but not limited to only these.
    c. *Evaluation* of frameworks/platforms/tools/plugins with appropriate datasets

    Most of the projects involve application of Natural Language Processing and/or Machine Learning methods/tools
3. **Projects in 1 and 2 but for Mobile Platforms**
4. **Projects in 1 and 2 but for Games/Game Engines/Game Platforms**

Computing for Education/Human Computer Interaction

5. **Skill Labs -** How do you design labs for 144 Skill Sector Councils as listed in National Skill Development Corporation? [https://www.nsdcindia.org] **-** Involves design of games or appropriate technologies to teach/learn skills and to automate the design and development of these technologies themselves.
6. **Analysis and Design of Gamification Engine**
    The goal is to analyze existing game/gamification engines to improve the platform through plugins
7. **Design of Technologies for Museums**
8. **Design of Technologies for Zoos**
    Both projects 7 and 8 involve design of technologies [such as games/Augmented Reality] to facilitate learning or activities in Museums and Zoos. For example, assuming there are about 100+ museums and 100+ zoos in the country. How do you design technologies for them is a computer science challenge.

Generally, the quest is:

#1 What kind of **[A* Novel Technology]** helps in teaching/learning/assessing [B* Subject/Topic] based on [C* Theory] in [D* Context] Improves [E* Skills]?, where A, B, C, D, E have to be decided based on mutual interest.

#2 How to automate the design and development of [A* Novel Technology]? while improving quality?

Generic Resources

1. Learn how to read and write research papers http://bit.ly/2ijsTW0 and http://bit.ly/2iPJDHZ.
2. Two good ways to find research papers is to find the best conferences/journals in that area [Look at attached note] and search for good professors/researchers in that area and see what kind of work are they doing? For example, for software engineering, ICSE, FSE, ASE, MSR and so on are good ones. For Human Computer Interaction, CHI and so on..
3. Two ways... Google Scholar [scholar.google.com] or Search DBLP [dblp.uni-trier.de] with names of researchers/conferences
4. Look for research projects at good places like MIT, Stanford, CMU, UW, Illinois and so on mainly in SOFTWARE ENGINEERING, HCI and Learning Technologies. Some persons that you should look for are: Andrew J Ko, Brad Myers, Mei Nagappan, Shane Mcintosh, Martin Robilard, Nenad Medvidovic, Thomas Zimmerman and so on.
5. Desirability and Feasibility [Be specific and concrete!] - Two important things as discussed in the class. One are of interest is Empirical Software Engineering that has both of these traits - Look at this paper
6. http://bit.ly/2jbrDUO
7. You must develop and implement the proposed solution using some programming languages and technologies. So learn that! Evolve your Github repositories
8. Learn some basic programming technologies like JS, Python, HTML5 and so on [May be tutorialspoint or something but definitely do a pet project]
9. Collaborate! May be as a first step, you can take projects in teams
10. Nothing will work without deadlines

Related Research Groups - Collaboration with both academia and industry

- http://sail.cs.queensu.ca
- https://cs.uwaterloo.ca/~m2nagapp/publications/index.html
- http://menzies.us/
- http://www.ing.unisannio.it/mdipenta/
- https://sites.google.com/site/asergrp/
- http://thesegalgroup.org/
- http://web.cs.ucla.edu/~miryung/
- http://ctreude.ca/publications/
- https://se.ewi.tudelft.nl/

Top Conferences and Journals

- Software Engineering - ACM SIGSOFT
    - ICSE, FSE, ASE
    - MSR, EASE, ESEM, ICSME
    - ICSA, SPLASH, RE
    - ICPC, SANER, SAC, EuropPLoP, ISEC
    - Journals: TSE, TOSEM, EMSE, JSS, IST
- Educational Technologies
    - SIGCSE, ITiCSE, ICER
    - AIED, ICALT, L@S, LAK
    - Journals: Computers & Education, TLT, BJET, ETRD, Smart Learning Environments, TOCE, IJAIED
- Human-Computer interaction - ACM SIGCHI

- CHI, CSCW, EICS, UIST, IUI, UMAP
- Journals: TOCHI, Computers in Human Behaviour

Resources

- Tao Xie's Advice Collection http://taoxie.cs.illinois.edu/advice.htm - One stop resource for most of the stuff!
- How to read and write a research paper?
  - http://taoxie.cs.illinois.edu/publications/writepapers.pdf
  - https://www.cs.cmu.edu/~Compose/shaw-icse03.pdf
- Do a crash course on
  - Research Methods [Not more than a week]
  - Technical Writing [Not more than a week]
- We will use Github, LateX, Mendeley/Zotero for managing bibliography and so on!
- Continuously improve your technical skills and share with the group, you should update about any new tool you are using for your research work for others to benefit here

  Note: Don't get bogged down by the word research, that's nothing but problem solving with novel techniques and for unsolved problems, which you do everyday!

Evaluation Criteria: [from ICSE]

- **Soundness:** How well the paper's contributions are supported by rigorous application of appropriate research methods,
- **Significance:** The extent to which the paper's contributions are novel, original, and important, with respect to the existing body of knowledge,
- **Verifiability:** Whether the paper includes sufficient information to support independent verification or replication of the paper's claimed contributions, such as data sources, scripts, code, data and so on
- **Presentation:** Whether the paper's quality of writing meets the high standards of ICSE, including clear descriptions and explanations, adequate use of the English language, absence of major ambiguity, clearly readable figures and tables, and adherence to the formatting instructions.

Empirical Study Project:

1. **Step 0 - Weekly Status Report**
   a. WSR
   b. Github link of code
   c. Overleaf document of the paper
2. **Step 1 - Background reading material**
   a. Ensure that you read the short and crisp "Elements of Style" http://www.jlakes.org/ch/web/The-elements-of-style.pdf
   b. Tao Xie's how to write papers
      i. http://taoxie.cs.illinois.edu/publications/writepapers.pdf
   c. Learn the process of Literature Review
      i. The only way to know what we are doing is good, is by knowing the literature well and proposed a solution better than existing work.
      ii. https://www.youtube.com/watch?v=WUErib-fXV0
      iii. https://www.youtube.com/watch?v=-ny_EUJXHHs
      iv. https://userpages.uni-koblenz.de/~laemmel/esecourse/slides/slr.pdf
   d. Tutorial on conducting empirical studies

   i. http://www.cs.toronto.edu/~sme/papers/2007/SelectingEmpiricalMethods.pdf
   ii. http://qse.ifs.tuwien.ac.at/wp-content/uploads/2007_CTU_LECT_EMSE.pdf
   iii. https://www.monperrus.net/martin/introduction-to-empirical-software-engineering.pdf
   iv. There is also a book in the library Empirical Methods in Software Engineering. In case if you require more details, lectures in http://pased.soccerlab.polymtl.ca/schedule.php

e. Learn basic mathematical foundations for empirical analysis such as probability and statistics, linear algebra, Natural Language Processing and AI/ML techniques. You will know what exactly to learn after Step 2.

3. **Step 2 - Literature Review**
   a. Browse through related papers and identify relevant papers [by reading and analysing abstracts and conclusions] of conferences and journals such as ICSE, FSE, ICSME, MSR, ESE and so on!
   b. Select related papers or empirical studies [about 10] and identify the Structure of the paper and argument, Research Questions, Data Sources, Techniques, Research Method and so on. Two ways... Google Scholar [scholar.google.com] or Search DBLP [dblp.uni-trier.de] with names of researchers/conferences and looking at recent papers with more citations, and digging through their references
   c. The outcome of this step is an excel sheet as in https://bit.ly/2SsvYt1 or https://bit.ly/2GGZI3l but can have different column headings depending on the research work and the write-up for literature review. Trust me! This the key step that decides whether the work you do is acceptable to the global community or not!
   d. Another outcome at this stage is a draft structure of the paper with all sections, detailing what to write in each of the sections.
4. **Step 3 - Experiment Design**
   a. Decide the precise study to be done - Two key criteria - Is it interesting problem? Is it unsolved? What is the novelty? What are the pain points in existing work? How your proposed work will address the challenges? How you propose to validate the work? Again, go back to literature to see how existing research work or papers have answered each of these questions, because it is the community who will evaluate your work. Assume if the best researchers and conferences/journals review your work, will they accept it?
   b. Refine the basic structure of the paper, RQs, data sources, techniques, expected results
   c. Frame Research Questions
      i. Write a draft of questions based on prior empirical studies, tutorial and literature review
5. **Step 4 - Conduct Experiment**
   a. Decide and Prepare Data Sources - either existing relevant dataset or create your own
   b. Quickly learn techniques required to conduct the study such as NLP/AI/ML depending on the empirical study
   c. Conduct the experiment - While conducting the experiment, ensure that you are taking care of all validity and bias concerns, expressed as Threats to Validity in the literature.
6. **Step 5 - Analyze and Produce Results**
   a. Critically analyze the results from the experiment with respect to state-of-the-art, and document the results
   b. This is a critical step as any claims we make as part of our research must be validated in this step
7. **Step 6 - Revise the experiment based on results**
   a. Revise the entire draft paper, methods, RQs, and so on based on the inital results
8. **Step 0-6 -** This is a continuous step to be done throughout from the beginning. Write the Empirical Study following the same structure of the conference or journal. Either IEEE/ACM, https://icsme2019.github.io/formatting_guidelines.html, usually 10 pages + 2 pages of references

Papers and Reviews: For every paper submitted, there should be a folder with the following details!

1. Create a folder in the format : (This folder will be considered head folder for each paper) LastNameofFirstAuthor_VenueSubmittedandYear_PaperTitle.

   Ex: Veda_ICSE2020_Title

2. Add the pdf version of the submitted paper in this folder.
3. Add the source files including tex, bib and other files, and they should follow the same naming convention
4. Add the tool repository source in the github folder along with readme and guidelines for others to use it!
5. Add a document and store all the reviews. Name it as LastNameofFirstAuthor_VenueSubmittedandYear_PaperTitle_Reviews
6. Add a sheet containing reviews, corresponding possible resolutions. Name this sheet as LastNameofFirstAuthor_VenueSubmittedandYear_PaperTitle_Resolutions.