National Resource for Network Biology: Student Profile

Shaik Asifullah developed a distributed computing framework for dynamical modeling of biochemical reaction networks with the help of his mentor J Kyle Medley. His framework allows optimizing stochastic biological models by allowing the optimization workload to be split up among computers in a cluster or a cloud platform. Shaik implemented this method by using Apache Spark to add distributed computing capability to the Tellurium project, a collection of Python packages for dynamical modeling in biology. His contributions have been integrated into Tellurium and are now part of the codebase.

- **Project Blog:** https://medium.com/@s.asifullah7/a-never-ending-journey-58b33da05409
- **Mentor Quote:** "I was very fortunate to get someone as hard-working and dedicated as Shaik to work on this project. Shaik took on a very challenging task - he had to add distributed computing capability to a collection of esoteric biological modeling packages. His self-reliance, autodidacticism, and eagerness to learn new things all helped propel his project to success." - *Kyle Medley*

What was your school / major during Google Summer of Code?

I participated in GSoC while pursuing a master's degree in Psychology at Dr. B. R. Ambedkar Open University (BRAOU), Kadapa

How did you find out about Google Summer of Code?

I did my graduation in Computer Science from BITS Pilani, Goa and many of my classmates, including BITS alumni, participated in GSoC. It's a way to get practical experience, sort of like an internship. This year, I decided to participate after getting inspired by my cousin, who also got selected for GSoC. Some of the classmates I mentioned also worked on projects through NRNB and had good experiences with their projects and mentors, which helped me narrow down which organization I wanted to participate through.

What factors helped you decide on a GSoC project?

I had recently completed a project using Twitter to forecast the results of the U.S. presidential election. In that project, I picked up a great deal of distributed computing skills, and I wanted to leverage those skills for GSoC. I also have a very unique research background - for my master's thesis, I combined "Big Data"-era technologies with social and behavioral sciences, and I wanted to see if I could expand my research even further. When I saw the project pitch for utilizing distributed computing in biological modeling, I knew it was the perfect fit.

What was your GSoC experience like? How did it compare to your expectations?

I never realized how supportive a free / open-source community could be. This was my first time contributing to an open-source project, and I didn't know what to expect beforehand, but the community was very helpful and encouraging. I really enjoyed learning about a new project and asking questions. I also got chance to get involved in presenting a research poster at Beacon 2017. It was a great learning experience.

What are your future career plans? What role does free / open-source software play in your work?

I really like research. I'd like to complete research-based master's degrees in both psychology and computer science, and possibly pursue a PhD after that. I think that now is a really exciting time to be getting into computing, because we have access to data and computing resources we never had before, and we can use these resources to make novel predictions. For example, in my previous project, we were able to successfully predict the results of the U.S. presidential election using data from multiple social media sources like Twitter, Blog Posts & News Channels and performing Sentiment Analysis on the dataset. Thanks to my GSoC project, I have extended the same high-throughput tools to biological modeling.

What are some of the features you implemented in Tellurium?

In biological models, there are usually a lot of unknowns which can't be measured directly. Also, there is intrinsic randomness inherent in each cell and any measurement data typically contains a lot of noise. This poses a problem for modeling because the randomness and noise makes it hard to tell whether a model is accurate or not. In my project, I tried to address this in the following ways:

- I developed a method to estimate the unknown parameters in a model, from a given set of data, in the presence of noise and randomness. The models are intrinsically *stochastic* (their output is random), so I had to do a lot of statistical sampling do get the answer to converge. This is where I used distributed computing. With Apache Spark, we can utilize the cloud or a cluster of computers to speed up the sampling process.
- A distributed stochastic method which runs computations in multiple executor nodes in parallel and then computing metrics for the parallel runs
- Distributed Parameter Scanning which allows users to perform parameter scanning of multiple models in parallel
- Sensitivity analysis of multiple parameters of the model(s) with Apache Spark