Sai Teja Pratap Suram

Skills Summary

- Systems: extensively worked with ElasticSearch, Redis, Aerospike, Hadoop, MySQL, AWS and familiar with Kafka, Spark, InfluxDb
- Languages: proficient in java, python, c++ and familiar with go, javascript

Education

- Bachelors in **Computer Science** from **Indian Institute of Technology, Bombay** (2009-2013)
- Worked in **Bing Data mining team** (Microsoft, Hyderabad) as **Software Engineer Intern**
- Secured Country (India) Rank of **57 among 300,000** students (99.98 percentile) in IIT-JEE 2009

Professional Experience

Software Engineer at Google Inc (Mountain View, CA) (from Jun 16)

I Work on Data Center Software.

Software Engineer at Relcy Inc (Mountain View, CA) (Apr 15 - May 16)

Relcy is a full fledged search engine (with 9 backend engineers) focussing on entity centric search. It has a knowledge graph with ~200 million entity objects (local businesses, people, movies, TV shows, songs). I worked on almost all components of Relcy. Some of my work can be found on Relcy Engineering Blog : Link to Blog

- **Migration of Primary Key Value Store**
  - Proposed, researched and completed migration from **Redis** to **Aerospike** as the primary key value store (3x capacity; same hardware; clustered solution)
  - Implemented a layer on top of aerospike data model to have versioned tables
  - Built custom metrics for aerospike and Amazon CloudWatch for alerting
- **Serving System**
  - Did huge code refactorings in the serving system to support new features and improve performance.
  - Improved **90 %ile search response time from 2 secs to 0.6 secs** and **99 %ile from 30 secs to 1 secs** by identifying redundant code paths, threading issues, duplicate db calls, tweaking db connection settings.
  - Worked with client engineers (IOS/Android) on extension to existing schema to support new features
- **Query Execution Framework:** Created framework for executing **chained structured queries** which converts Query Interpretations to **Elastic Search** Queries and executes them
- **Analytics Framework :** Worked on designing the overall framework and mentored a new team member to build the first version (using **Kafka**, **Spark**, InfluxDb & Mysql**)**
- **Data Mining/Inference**
  - Integrated **Wikipedia data** (from dbpedia) into Relcy Knowledge Graph to serve 'topic' entities. Trimmed down **Wikipedia category graph** from ~1 million to a few dozen and used these for type inferencing.
  - Created **Streets Dataset** from Relcy Entities. Enriched Relcy **City Dataset** with Wikipedia information.
  - Worked on Web vs Entity Classifier which decides whether to show Relcy entities or web results on top. Used gradient boosting decision trees classifier
  - Created pipeline for alias creation for celebrity entities in Relcy Knowledge graph (ex: when people search for 'Nolan', they are very likely looking for 'Christopher Nolan')
- Worked with Relcy crawlers to create a crawler aimed at websites for which the exact structure is known

Software Engineer at Rocket Fuel Inc (Redwood City, CA) (Nov 13 - Mar 15)

Rocket Fuel handles ~100 billion ad bid requests per day from various ad exchanges and bids on their inventory to show ads. I worked in the serving side of Rocket Fuel

- Changed the core bidding logic to **gracefully handle bid response timeouts** (instead of evaluating all ads and potentially timing out, serve the best ad by not evaluating some ads if a timeout is foreseen)
- Built **Ad Campaign diagoniser** to diagnose ad campaigns & give potential reasons why they may not deliver
- **Reporting pipeline**: Stabilized & enhanced bidding reports system (reports sent by ad exchanges on Rocket Fuel's summary and vice versa). Built the pipeline to ingest publisher blocked sites into Rocket Fuel's system
- **Exchange Integrations:** worked on integrating ad exchanges (Tremor, Rubicon) with Rocket Fuel.

Pet Projects & Hobbies

- I built my personal website + blog (https://yesteapea.com) with a combination of static and dynamic content using nginx, jekyll, python flask, golang on an aws virtual machine. This blog post explains how I built it.
- I built a dictionary plugin for slack hosted on my website (link to plugin page)
- I enjoy long bike rides, short runs and small hikes