

Rishabh Misra

✉ rishabhmisra1994@gmail.com

☎ +1 (650)-686-9605

💼 [linkedin.com/in/misrarishabh](https://www.linkedin.com/in/misrarishabh)

💻 [rishabhmisra.github.io](https://github.com/rishabhmisra)

EDUCATION

M.S. Computer Science	University of California San Diego	GPA: 3.93/4	Sep 2016 – Jun 2018
B.E. Computer Engineering	Thapar University, India	GPA: 9.88/10	Jul 2011 – Jul 2015

INDUSTRY EXPERIENCE

- **SDE Intern @ Amazon.com, Seattle, Washington** Jun 2017 – Sep 2017
 - I interned in the **DataForge** team which provides a **platform for running Big Data operational workloads** consistently within service level agreement. I worked towards **designing and implementing**:
 - Support for **primary key constraint** and **batch insert/update** while ensuring consistent reads in Hive using **type-2 tables** and **multi-version concurrency control** concepts.
 - Support for **ACID properties in Hive**.
 - Support for non-blocking **compaction** (carefully discarding old data) to keep read operations efficient.
 - **Technologies**: Java | Hive | DynamoDB
- **Member Technical @ Arcesium India Pvt. Ltd., India** Jul 2015 – Jul 2016
 - Worked for the Straight Through Processing team. Important responsibilities:
 - Added support for **self-sanitization, self-recovery** and **fault tolerance** in the new infrastructure.
 - Added **self-aware trigger mechanism** for EOD trade files, **minimizing** data completeness issues by **30%**.
 - **Technologies**: Java | Spring Framework | MyBatis | Microsoft SQL Server

RESEARCH EXPERIENCE AND PROJECTS

- **Student Researcher under Prof. Julian McAuley @ UCSD** Apr 2017 – Present
 - **Spoiler Detection** Python | Selenium
 - To **detect spoilers** in **book/movie reviews** and produce some **interesting linguistic insights** on what constitute spoilers, developing a **probabilistic graphical model with integrated language model**.
 - Data scraped from two popular websites, then cleaned, pre-processed and explored.
 - **Product Size Recommendation** – People always worry whether the **product** they're **buying online** would **fit**.
 - Aims to **predict the fit of clothes** using a **metric learning model**, thus improving recommendation.
 - **Improves** upon an algorithm developed by [Amazon](#) up to **13%**.
 - Data scraped from two clothing websites with fit feedback, then cleaned, pre-processed and explored.
- **Research Intern @ Indian Institute of Technology, Madras** May – Jul 2014, Dec 2014 – May 2015
 - **Scalable Bayesian Matrix Factorization algorithm**: reduces the cubic time complexity of existing Bayesian matrix factorization algorithm to linear. (C++ | Python | Matlab) (Link: goo.gl/ou2B7f)
 - **Scalable Variational Bayesian Factorization Machines**: Supplements the existing framework with scalable alternative that gives state-of-the-art performance. (C++ | Python | Matlab) (Link: goo.gl/nH59G4)
 - **Collaborative Tweet Recommendation**: Used Parametric Matrix Factorization to efficiently recommend relevant tweets to users. (C++ | Python)

PROJECTS

- **Jointly Modeling Aspects, Ratings and Sentiments with Temporal Dynamics** Python
 - We **jointly model aspects** of the products, **user sentiment** on products, associated **ratings** and **temporal information** in a **probabilistic graphical model** to **predict** the review **ratings**. For interpretability, model **produces insights** on the various aspects of products and user sentiment on them which explains the rating.
 - **Improves** upon the original method by **1%** and **provides insight** into **users' preference change over time**.
- **Review Ranking and Recommendation on Ciao Product Dataset** Java
 - Reviews ranked and recommended by **optimizing** the **Bayesian Personalized Ranking** measure on **biased Matrix Factorization** and **biased Tensor Factorization** models.
- **Hierarchical Attention Network for Rating Prediction** Python | Tensorflow
 - Implemented an **RNN with attention mechanism** for **rating prediction using product reviews**. Attention mechanism allows the **RNN** to **focus on sentences** which **best explain the rating** given to an item.
- **An Ensemble of CNNs for Traffic Lights Recognition** Python | Keras framework
 - **Ensemble** of custom built **CNNs** trained on **Nexar traffic lights challenge dataset** while **ensuring small model size** which allows for a **quick training** even with **scarce computational resources**.
- **Hotel Recommendation System** Python | Scikit-learn
 - **Recommender system** trained on Expedia Hotel Recommendation Dataset to **recommend top 5 hotel clusters** to users, built using **ensemble** of **Random Forest, Naïve Bayes, SGD classifier** and **XGBoost** models.