

TENGDA WANG

(412) 214-2825 | tengdaw@cs.cmu.edu | cs.cmu.edu/~tengdaw | linkedin.com/in/tengda-wang/

Education

Carnegie Mellon University

Pittsburgh, PA

Master of Computational Data Science, **3.89/4.0 GPA**

Dec 2024

- **Key Courses:** Machine Learning, Advanced NLP (PhD), Deep Learning Systems, Search Engine, Database Systems, Distributed Systems, Advanced Cloud Computing, Parallel Architecture And Programming, Computer Systems.

National University of Singapore

Singapore

Bachelor of Science (Honors) in Business Analytics, **4.87/5.0 GPA**

May 2021

- **Awards:** IMDA Excellence Prize (most outstanding graduate), Graduation Valedictorian, Dean's Lists

Skills

Programming Languages: Python, Java, Scala, SQL, C, C++, Scala, Javascript (AngularJS/Vue.js), Bash

Machine Learning/Data Science: PyTorch, TensorFlow, Scikit-Learn, Keras, NumPy, Pandas, Tableau

Data: Spark, Kafka, Hadoop MapReduce, Airflow, Hive, HDFS, Redis, Postgres

Cloud/DevOps: Docker, Kubernetes, Terraform, AWS (EC2, S3, Lambda, CWL, SageMaker), GCP, Azure

Professional Experience

TikTok

Bellevue, WA

Machine Learning Engineer Intern, E-commerce Recommendation Team

Jun 2024 - Aug 2024

- Spearheaded the incorporation of advertisement signals to the TikTok Shop recommendation system. Iterated on and deployed multiple **collaborative-filtering** based algorithms in **HiveSQL** and **C++** for efficient product **recall**.
- Optimized **two-tower embedding recall models** with multi-stage conversion objectives, focusing on feature engineering, model architectures, and data pipeline. Models were deployed online and brought **positive GMV impact**.

Shopee

Singapore

Machine Learning Engineer, Search Team

Sept 2021 - Apr 2023

- **Query-Category Relevance:** Improved **search quality** for million users by boosting relevant items in the search result. Achieved 92.4% offline training AUC, increased CTR by 3.71% (**A/B test**) and decreased bad case rate by 20.21% online.
- Pre-trained monolingual **BERT** models using a **masked language task** on item descriptions in 8 different languages, which improved performance of downstream tasks (NER, query rewrite etc) in both **feature-based** and **fine-tuning** fashion.
- Architected, implemented, and maintained an end-to-end distributed pipeline made up of a large data warehouse, an ETL feature-extraction pipeline, a model training and inference module, and a deployment layer utilizing Redis cache, that handles up to 100 TB data with thousands QPS.
- Collaborated with 5+ Product Managers and Product Ops across 8 regions to generate 10 million rows of human-labeled data for model training, and increased model offline metrics by 44.2%.

Bank of America Merrill Lynch

Singapore

Software Engineer Intern, Global Markets Tech Team

Jun 2020 - Aug 2020

- Formulated workflows and created multiple **full-stack web applications** including frontend (**AngularJS**), backend (**Scala**), and unit-testing (**ScalaTest**) to help clients manage portfolios. Worked closely with product side to ensure smooth **UI/UX**.

Selected Projects

- Needle (2024), Developed a PyTorch-like deep learning framework from scratch with support for common **neural layers** (**CNN**, **LSTM**, **Transformers** etc), **autodiff**, **dataloader**, and **NDArray** speed up on CPU/GPU backend. (NumPy, C++,CUDA)
- QueryEval (2024), Built a full-fledged search engine on top of **Apache Lucene**. It handles **query parsing**, initial retrieval via **BM25** and **Indri**, and improves relevance with **pseudo relevance feedback** and **learning-to-rank** based reranking. (Python)
- Bachelor Thesis (2021), Studied **neural abstractive summarization** techniques (e.g. **Transformers**, **Seq2Seq models**) to automatically generate hospital discharge summaries in electronic health records. (PyTorch)
- BusTub (2024), Extended the functionality of a RDBMS by implementing a efficient buffer pool manager with LRU-K eviction policy, a disk-based concurrent B+ tree index with fine-grained locking, an execution engine with query optimization capabilities, and a multi-version concurrency control protocol (MVCC) for database transactions. (C++)
- Distributed Proxy (2024), Designed and coded a distributed proxy server that supports whole-file caching and LRU eviction. The proxy uses Java RMI as the underlying RPC protocol, and leverages check-on-use techniques to ensure cache consistency in open-close session semantics similar to the Andrew File System (AFS). (Java)