Mingchen Li

Researcher, Division of Computational Health Sciences, University of Minnesota Twin Cities

Email: li003378@umn.edu
Mobile: 470-815-7917

Please Click My Blog
Please Click My Github

SUMMARY

My principal research interests lie in the area of **Natural Language Processing**, **Machine Learning**, **Deep Learning** and **Data Mining**.

Education

- **Georgia State University (GSU)** — ATL, US
  M.S. in Computer Science, GPA: 4.0/4.3 — *1.2021-12.2022*
- **Qufu Normal University (QFNU)** — Shandong, China
  B.S. in Computer Science, GPA: 3.6/4.3 — *2015-2019*

Experience and Projects

- **University of Minnesota Twin Cities** — 01.2022-Present
  Researcher
  - **1**: Health NLP
- **Georgia state university-ARCTIC** — 08.2022-12.2022
  Research Assistant
  - **1**: Created the **Spatio-temporal model** by using the **attention network** for pandemic prediction COVID 19. Input the graph embedding to gated recurrent unit (GRU) network to learn temporal features.
- **Prescio Consulting** — 06.2022-08.2022
  Internship
  - **Outcome**: This project is used to teach the client how to understand the interpretability of the ML model easily.
  - **1**: Compared the interpretability (by **SHAP**, etc) with different machine learning frameworks (**PyCaret and PIML**) under different models: **KNN, SVM, Decision Tree, Cluster, Random Forest**, etc.
  - **2**: **Data cleaning**, **data normalization** are done in the data progress. The data is the real-world **financial dataset**.
- **Natural Language Processing Center, Virginia Tech** — 01.2022-12.2022
  Research Assistant
  - **1**: Designed an **encoder-decoder language model** by using the **prompt tuning** and **structure information** to **extract** the latent **entities** and **relations** in the **Open Corpus** OpenPI.
- **Computer Science, Georgia State University (GSU)** — 01.2021-05.2021
  Graduate Research Assistant
  - **Outcome**: Get the **SOTA performance** in two complex KBQA datasets, MetaQA and WebQuestionsSP, especially in MetaQA-hop1, this model can get **99.1 Hit@1 values**, the best baseline has a 1.6-point lower accuracy than that of our method. More importantly, by our denoise method, **ranking speed has been greatly improved**. Our **classification model** is **beyond BERT 2.8 points in MetaQA Hop1**.
  - **1**: In the task of **question answering** over knowledge graph, I proposed a new method to denoise the candidate query graph and used a novel ranking model to get the predicted query graph.
  - **2**: A novel classifier **StructureBERT** is proposed to predict the Semantic Structure for each question. A novel BERT model is proposed to **query graph ranking**. This work is accepted by **COLING 2022**.

Publications and Patents

8). **Mingchen Li**, Yang Ye, etc. **W-PROCER: Weighted Prototypical Contrastive Learning for Medical Few-Shot Named Entity Recognition**. Under Review 2023. [PDF], [Code].
7). **Mingchen Li**, Lifu Huang. **Understand the Dynamic World: An End-to-End Knowledge Informed Framework for Open Domain Entity State Tracking**. SIGIR 2023. [PDF], [Code].
6). **Mingchen Li**, Junfan Chen, etc. **A Hierarchical N-Gram Framework for Zero-Shot Link Prediction**. EMNLP 2022 Findings. [PDF], [Code]
5). **Mingchen Li**, Shihao Ji. **Semantic Structure based Query Graph Prediction for Question Answering over Knowledge Graph**. COLING 2022. [PDF], [Code].
4). **Mingchen Li**, Zili Zhou, etc. **Multi-Fusion Chinese WordNet (MCW): Compound of Machine Learning and Manual Correction**. CICLing 2019. [PDF].
3). **Mingchen Li**, Zili Zhou, etc. **Solving the Chinese Physical Problem Based on Deep Learning and Knowledge Graph**. ICITE 2019. [PDF].
2). Yanna Wang, Zili Zhou, **Mingchen Li**, etc. **An intelligent collection system for testing paper**. Publication number: CN107908752A. (**1st** student author).
1). Zili Zhou, Yanna Wang, Jinghu Zhang, Ning Zhang, **Mingchen Li**, etc. **An intelligent hardware control method driven by knowledge graph**. Publication number: CN107272521B. (**2st** student author).

Skills Summary

- **Languages**: Python, Java, SQL(Oracle), Unix, Linux, JSON, C++, C
- **Frameworks**: TensorFlow, Pytorch, MySQL, MongoDB, Neo4j, Transformers, Scikit-Learn, Pandas, PyCaret, PiML, Vaex