# Xiao Qin

773-290-3230 | xiaoqin2026.1@u.northwestern.edu | LinkedIn-spencer0220

## Professional Summary

Highly skilled Machine Learning Engineer and Data Scientist with diverse experience in large-scale data processing, AI-driven systems, cloud computing, particularly focus on large language model, data analysis, and RAG techniques

## Experience

**CO-Founder & Full-Stack Engineer** June 2024 – Present
*AccunoteAI - https://accunoteai.com* *Toronto, Canada*

- **Fine-tuned meeting speech understanding LLMs** and **designed complex system** to implement a comprehensive AI assistant for meeting notes. Developed full-stack web applications using **Fastapi**, **MySQL**, **Nginx**, **Docker** and **AWS** services (Including EC2, Route53, SageMaker, Lambda, etc.)
- Applied **keyword and semantic matching** to align numerical data with meeting records, replacing incorrect figures based on predefined rules. Reduced numerical error rate by **30**% compared to direct GPT-4 generation
- Introduced **breadth-first reasoning** with LLMs to correct ASR output errors related to entity names and technical terms, utilizing **knowledge graph** built from open-source company information datasets, achieving over **80**% accuracy in correcting misidentified terms

**NLP Engineer, Intern** Mar 2024 – June 2024
*Giant Interactive Group Inc.* *Shanghai, China*

- Conducted secondary development of **Qanything** for document understanding, and used LLM to implement a **JSON-based agent** that generates data visualizations for users
- Generated high-quality role-playing dialogue datasets using GPT4 to distill data from **web scraping** and open-source. Utilized **MiniHash** and **Spark** for efficient data cleaning and deduplication

**NLP Engineer** July 2023 – Mar 2024
*Zhejiang Engineering Digital Technology Research Institute* *Zhejiang, China*

- Fine-tuned **NTK-aware Scaled RoPE** Baichuan2-7B on generated instruction datasets from specifications document to implement a document QA chat-bot
- Implemented two **ensemble search** methods in **RAG** pipeline. Improved retrieval recall by fine-tuning the BGE embedding model and configuring the corresponding **keyword mapping dictionaries** for **Milvus** partitions as filters. Used **RAGAS** and **LlamaIndex** to evaluate RAG indicators, showing a 6% performance improvement
- Used **self-information** for **text compression** to tackle the challenge of LLMs struggling with long prompts

## Projects

**Traffic Accident Analysis Algorithm for Incomplete Data** July 2022 - Mar 2023

- Reproduced the **RelGAN**, **CTGAN**, and **TVAE** models on a small-scale traffic accident dataset containing 1,000 records, generating 3,000 high-quality synthetic samples. Optimized the **conditional generator** of CTGAN to align the sampling probability with the feature distribution of the real data, reducing the distribution discrepancy of the generated data by **20**%
- Trained a **transfer learning cost-sensitive SVM** model using both the generated synthetic data and the original data. Analyzed the top 10 most influential factors based on the SVM parameters. Compared to the SVM model trained without the generated data, achieved a **8**% increase in overall accuracy and a **15**% improvement in the recognition rate of minority classes

**Bert Based News Text Classification** Mar 2021 – Oct 2021

- Perturbed the original parameters of the **pre-trained Bert** model through the noise matrix, and search for optimal boundary values of the uniform distribution function and noise intensity in experiments
- Defined a plugin-style class with attack and restore methods to implement **FGSM** to do adversarial training
- Incorporated the crawler-expanded data set, improved the classification accuracy by **4**% compared to the baseline

## Education

**Northwestern University** Evanston, IL
*Master of Science in Artificial Intelligence* *Sep 2024 – Dec 2025*
**Hohai University** Jiangsu, China
*Bachelor of Engineering, Computer Science* *Sep 2019 – June 2023*

## Technical Skills

**Languages**: Python, Java, C, C++, SQL, JavaScript/HTML/CSS, Matlab
**Frameworks**: Pytorch, DeepSeek, Langchain, Pandas, Sklearn, React, FastAPI, SpringBoot, Spark, Git, Docker, AWS