CV Sorting using LLMs — Report

Title

CV Sorting using LLMs — Capstone — Sirsha Dattasarma

Abstract

(brief summary of aims and outcomes)

1. Introduction & Scope

- Problem statement
- Intended users (recruiters, HR)

2. Models & Tools

- Resume parsing: pdfminer.six, python-docx (or pyresparser)
- Embeddings: sentence-transformers (all-MiniLM-L6-v2)
- Orchestration: custom scripts, Streamlit UI
- Optional: LLM scoring (OpenAI/local)

3. Pipeline Architecture

- Data ingestion -> parsing -> embeddings -> scoring -> ranking -> UI (Include architecture diagram - draw in slide or paste an image)

4. Implementation Details

- Key modules: src/parse_resumes.py, src/parse_jd.py, src/embeddings.py, src/scorer.py, src/ranker.py, src/app_streamlit.py
- How anonymization handled
- Dockerfile & reproducibility

5. Evaluation

- Dataset: number of resumes (N), JD used
- Metrics: Precision@k, Average Precision (AP), MAP, Kendall Tau
- Results: paste metrics from results/eval_metrics.json