Automated Resume Parsing Using Named Entity Recognition (NER) with NLP

Introduction

Named Entity Recognition (NER) is a very important Natural Language Processing (NLP) technique that helps in extraction of relevant information from text. One of its biggest uses is automated resume parsing this automates the process of working through resumes and gives an organization a way to quickly sort through great masses of them to find things like:

- Personal Information (Name, contact details)
- Education (Degrees, institutions)
- Work Experience (Job titles, company names)
- Skills (Technical proficiencies, languages)

The Resume NER Dataset contains 36 entity categories over resumes and presents the possibilities of creating highly accurate NER models for the research. The use of NER for resume parsing can help the recruiters of the organizations to automate the talent acquisition, increased efficiency and facilitate the decision making of the recruiting department from the recruitment side.

Dataset Link: https://github.com/vrundag91/Resume-Corpus-Dataset

Methodology

The implementation of Automated Resume Parsing using Named Entity Recognition (NER) with NLP goes through a structured pipeline of few major stages:

1. Data Collection and Preprocessing

- We use the Resume NER Dataset from GitHub with 36 entity categories for each resume.
- Data is preprocessed by:
  - Text Input: Taking in text, sentence, or paragraph information as input.
  - Standardizing text and removing uninformative words is called Lowercasing & Stopword Removal, or o.
  - Lemmatization/Stemming: Reducing words to their root forms.
  - Handling Special Characters: Cleaning special characters like formatting problems.

2. Named Entity Recognition (NER) Model Selection

To extract key resume entities, different NER models are considered:

- **Rule-Based Approaches**: Using handcrafted regex patterns for entity extraction.
- **Machine Learning-Based Approaches**:

    - **Conditional Random Fields (CRF)** or **Hidden Markov Models (HMM)** for sequence labeling.

- **Deep Learning-Based Approaches**:
    - **BiLSTM-CRF**: A combination of Bidirectional Long Short-Term Memory (BiLSTM) and CRF.
    - **Transformer-based Models**: **BERT, RoBERTa, or SpaCy's pre-trained NER models** fine-tuned on the dataset.

3. Model Training and Evaluation

- The dataset is split into **training, validation, and test sets** (e.g., 80-10-10).
- The selected model is trained using appropriate optimization techniques (e.g., Adam optimizer) and loss functions (e.g., categorical cross-entropy).
- **Evaluation Metrics**: Precision, Recall, and F1-score are used to measure model performance.

4. Post-Processing and Data Structuring

- Extracted entities are mapped to predefined categories (e.g., Name, Education, Skills).
- Data is structured into **JSON, CSV, or database format** for easy integration with HR systems.

5. Deployment and Integration

- The trained model is deployed using **Flask/FastAPI/Django** for API-based integration with applicant tracking systems (ATS).
- The system is tested on real-world resumes to refine performance.

6. Continuous Learning and Optimization

- The model is retrained periodically with updated resume datasets.
- Performance is optimized through **hyperparameter tuning**, **data augmentation**, and **active learning** strategies.