Project Report: Resume Extractor
Name: Zongdao Wen
Internship Batch: LISUM21

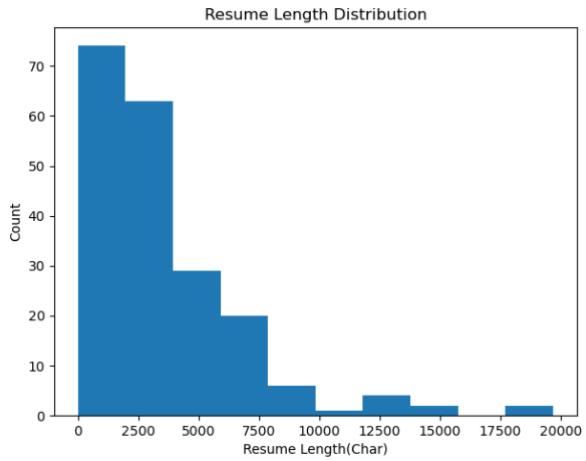
Project Overview

The Resume Extractor project aims to utilize Natural Language Processing (NLP) techniques, specifically Named Entity Recognition (NER), to classify and extract relevant information from resumes. The objective is to identify and categorize entities such as person names, college names, academics information, experiences, skill sets, and more from a given resume. The model is trained on a dataset of 200 annotated resumes.

Dataset and Entity Categories

The dataset consists of 200 resumes, with an average length of 2582 characters. The resumes are labeled with 10 entity categories:

1. Years of Experience
2. Graduation Year
3. Companies Worked At
4. College Name
5. Name
6. Designation
7. Location
8. Degree
9. Email Address
10. Skills

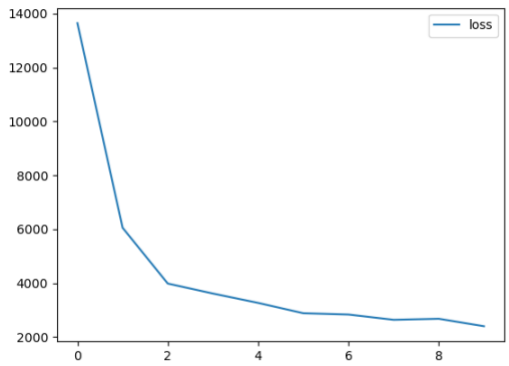


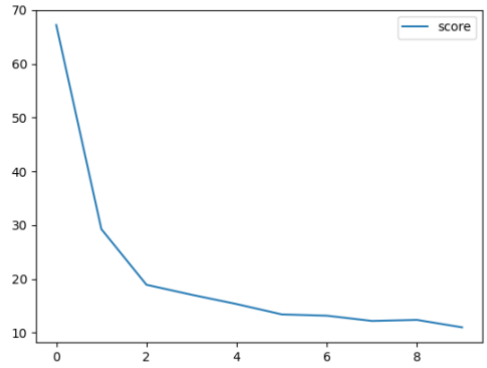
Data Preprocessing

Before training the model, the data is preprocessed to clean and tokenize the text. Stop words and punctuation are removed, and the text is tokenized into words. Additionally, the data is transformed into numerical representations suitable for NLP model training.

Model Training and Evaluation

The NER model is trained on the preprocessed dataset using machine learning algorithms. During training, the loss and score of the model are plotted to monitor its performance. The model is evaluated on a separate validation set to ensure its accuracy and generalization capabilities.

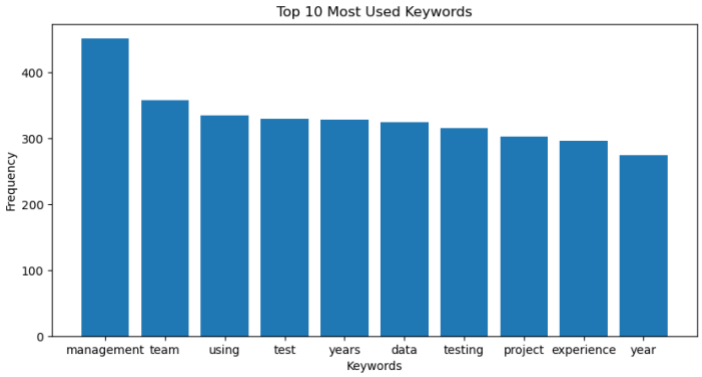




Top 10 Frequent Words

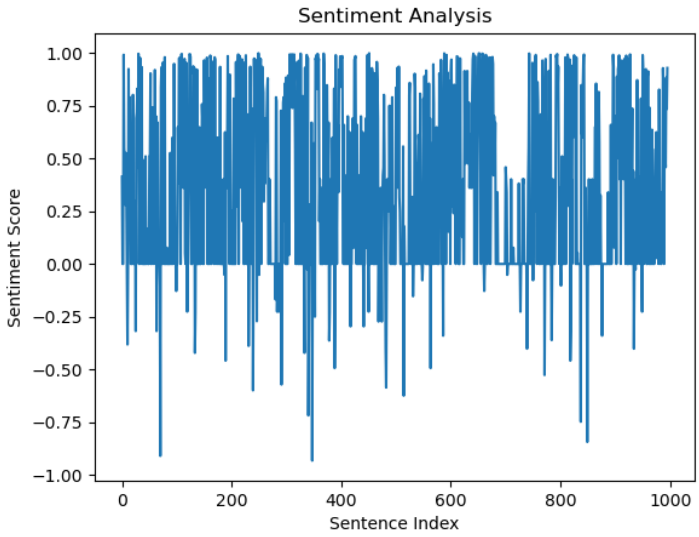
The analysis of the dataset revealed the top 10 most frequent words used in resumes, ranked from highest to lowest:

1. Management
2. Team
3. Using
4. Testing
5. Years
6. Data
7. Project
8. Experience
9. Year
10. Skill



Sentiment Analysis

An overall sentiment analysis of the resumes indicates a positive tone throughout the text, which is beneficial for potential employers looking for positive attributes in candidates.



Integration into Flask App

The trained NER model is integrated into a user-friendly Flask web application. The app allows users to input a PDF resume, and the model parses the resume to extract relevant information. The Flask app provides a seamless and efficient user experience, making it easy for employers to quickly identify key details in a candidate's resume.

Conclusion

The Resume Extractor project demonstrates the effective utilization of NER models in extracting and classifying entities from resumes. The integration into a Flask app streamlines the process of analyzing and understanding resume content, providing valuable insights for hiring decisions. The positive sentiment and predominant use of relevant nouns in the dataset further validate the effectiveness of the NER model in capturing essential information from resumes.

The Resume Extractor project holds significant potential in assisting recruiters and employers in identifying qualified candidates efficiently. As the model's accuracy improves with larger datasets and advancements in NLP, the application can become a valuable tool in talent acquisition and resume screening processes.