“Resume Parsing and Template Filling”

ANUSHTHA BAGERIA (IIT2020064)
6th SEMESTER, B-TECH INFORMATION TECHNOLOGY DEPARTMENT
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

***Abstract*—The project aims to develop a natural language processing (NLP) system to automate the process of parsing resumes and filling templates for job postings. The proposed system uses machine learning techniques to analyze the content of resumes and extract relevant information, such as name,email,companies worked at,location, education, work experience, and skills. NLP techniques, such as named entity recognition and keyword extraction, have been used to automate the process of extracting key information from applicant’s resume.**

I. Introduction

A. Introduction to Resume Parsing

In today’s fast-paced job market, companies receive a large number of job applications for each job posting. Manually screening and analyzing each resume can be a time-consuming and challenging task for recruiters. To overcome this challenge, resume parsing has become an increasingly popular solution.

A resume parsing is an analysis of the technology and processes involved in extracting relevant information from a job seeker’s resume. This report aims to provide an overview of how resume parsing works, the challenges and opportunities associated with it, and its impact on the recruitment process. This report will explore the techniques used to extract data from resumes with the help of NLP techniques.

The development of automated resume parsing systems has the potential to revolutionize the recruitment process by streamlining the initial screening and selection process. In this way, recruiters can focus on the most promising candidates and speed up the recruitment process, improving efficiency and productivity.

Overall, the increasing use of automated resume parsing highlights the importance of leveraging technology to optimize the recruitment process and find the best candidates for each job opening.

B. Goal

The goal of resume parsing using NLP is to extract relevant information from resumes and convert it into structured data that can be easily analyzed and used for various purposes such as recruitment, talent management, and HR analytics.

TABLE I
Literature Review Table

S.no	Author	year of publication	Paper Title	Observations
1.	Shubham Bhor, Vivek Gupta, Vishak Nair, Harish Shinde,Prof. Manasi S.Kulkarni5	2021	Resume Parser Using Natural Language Processing Techniques	Resumes will be ranked in order and Used NER.
2.	Nirali Bhaliya, Jay Gandhi, Dheeraj Kumar Singh	2020	NLP based Extraction of Relevant Resume using Machine Learning	Semantic mapping for limits and Parsing with lease limit.
3.	Papiya Das,Manjusha Pandey and Siddharth Swarup Rautaray	2018	A CV Parser Model using Entity Extraction Process and Big Data Tools	Convert unstructured resumes to structured,Used techniques like Extraction of Entity,POS tagging
4.	VINAYA RAMESH KUDATARKAR, MANJULA RAMANNAVAR, DR. NANDINI S.SIDNAL	2015	A Survey on Unstructured Text Analytics Approaches for Qualitative Evaluation of Resumes	unsupervised learning and ranks based on cosine similarity.
5.	Satyaki Sanyal,Souvik Hazra, Soumyashree Adhikary, Neelanjan Ghosh	2017	Resume Parser with Natural Language Processing	scrape keywords from different social networking sites including Stack Overflow, LinkedIn, etc

II. Literature Review

III. Problem Formulation

The current process of manually reviewing and extracting relevant information from job applicants’ resumes is time-consuming and prone to errors, leading to inefficiencies and delays in the recruitment process. There is a need to develop an automated system that can accurately parse and extract key information from resumes using NLP techniques to streamline the recruitment process and improve its efficiency. Biasing, Inconsistency, Time-consuming, Human-error, and Limited scope are a few problems that are faced during manual resume parsing. Specific challenges to be addressed while automating this process include:

- Unstructured data: Resumes are often unstructured and contain information in various formats such as bullet points, tables, and paragraphs. This can make it difficult to extract relevant information accurately.
- Identifying and extracting key information: Key information such as name, contact information, work experience, education, and skills accurately and efficiently, despite variations in how this information is presented.
- Ambiguity: Resumes can be ambiguous, and information may be incomplete or unclear. For example, a candidate may mention that they have ”experience in marketing,” but it may be unclear how much experience they have, what specific marketing skills they have, or in what context they gained that experience.

IV. Methodology

- Data preprocessing: It is an essential step in resume parsing, as it helps to clean and transform the raw text data extracted from resumes into a structured format that can be analyzed and used for further processing.
  - adding annotation in data
  - Removing whitespaces
  - Removing symbols
  - Removing special characters
- Named Entity Recognition (NER): It is a subtask of natural language processing (NLP) that involves identifying and classifying named entities in unstructured text data into predefined categories such as people, organizations, locations, dates, and other types of entities. The named entities can be proper nouns or noun phrases that refer to specific entities or concepts mentioned in the text.

  In resume parsing, NER algorithms use machine learning and statistical techniques to analyze the text data of resumes and identify the relevant named entities, such as the candidate's name, email address, phone number, work experience, education, skills, and other relevant information. By extracting these named entities, the parsed resume can be further processed and analyzed to match job requirements or to populate database fields for easier candidate evaluation. NER is a crucial component of resume parsing systems as it enables the accurate and automated extraction of important information from unstructured text data.

  NER is an essential component of many NLP applications, including resume parsing, chatbots, question-answering systems, and sentiment analysis.
- Template Filling: Finally, after NER we will move forward to fill out the designed template. The template helps us to find the data like name, job, company, skills, etc.

A. Dataset

The dataset I used is ResumeParsing available on kaggle.

About Dataset:
This dataset contains 2 folders named test and training. Test folder contains pdf and txt files for testing the model. And training model contains configuration file, train_data in json and pkl format.

Inside the json file:
It contains the annotated data which means text of the resume along with the entities for training our NER model.

The entities defined in the dataset were not up to the mark so there was a need to update the entities of the data. Therefore I have updated the entities with the help of website Annotator.

V. Result

The final score achieved was 0.69. In general, accuracy is not always a good measure of performance for NER models, as the datasets can be highly imbalanced, with many more non-entities than entities. Therefore, F1 score is often used as a more informative evaluation metric for NER models.

E: Epoch number

#: Batch number

LOSS TRANS: The loss of the transformer model during training. This is a measure of how well the model is fitting the training data.

LOSS NER: The loss of the named entity recognition (NER) component of the model during training. NER is a task in NLP that involves identifying and classifying named entities in text, such as people, organizations, and locations.

ENTS_F: The F1 score of the model on the NER task during training. The F1 score is a measure of the model's precision and recall, which combines both metrics into a single score.

ENTS_P: The precision of the model on the NER task during training. Precision is a measure of the model's ability to correctly identify positive examples (i.e., named entities) out of all the examples it identifies.

ENTS_R: The recall of the model on the NER task during training. Recall is a measure of the model's ability to identify all positive examples out of all the true positive examples that exist.

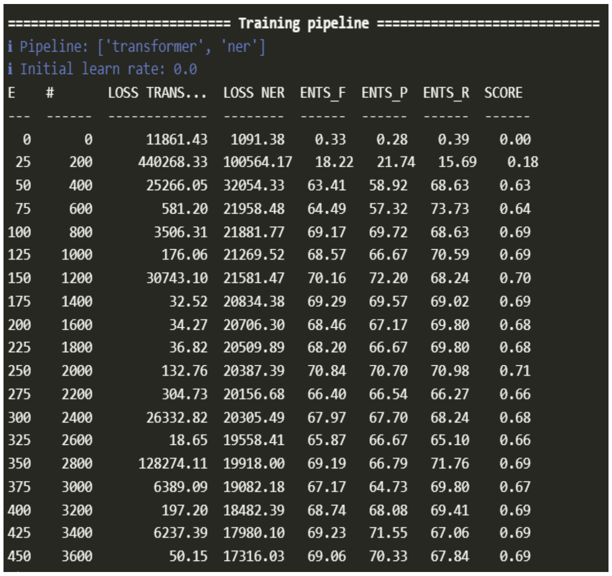
SCORE: The overall score of the model on the NER task during training. This score may be calculated using a combination of precision, recall, and other metrics, and it provides an overall measure of the model's performance on the task.

The github link of the project is Github-NLP_COURSE_PROJECT.

A. During Training

VI. Future Scope

The current model has been trained and tested on textual data, but there is potential for further development and



```
========================== Training pipeline ==========================
ℹ Pipeline: ['transformer', 'ner']
ℹ Initial learn rate: 0.0
E    #       LOSS TRANS...  LOSS NER  ENTS_F  ENTS_P  ENTS_R  SCORE
---  ------  -------------  --------  ------  ------  ------  ------
  0       0       11861.43   1091.38    0.33    0.28    0.39    0.00
 25     200      440268.33 100564.17   18.22   21.74   15.69    0.18
 50     400       25266.05  32054.33   63.41   58.92   68.63    0.63
 75     600         581.20  21958.48   64.49   57.32   73.73    0.64
100     800        3506.31  21881.77   69.17   69.72   68.63    0.69
125    1000         176.06  21269.52   68.57   66.67   70.59    0.69
150    1200       30743.10  21581.47   70.16   72.20   68.24    0.70
175    1400          32.52  20834.38   69.29   69.57   69.02    0.69
200    1600          34.27  20706.30   68.46   67.17   69.80    0.68
225    1800          36.82  20509.89   68.20   66.67   69.80    0.68
250    2000         132.76  20387.39   70.84   70.70   70.98    0.71
275    2200         304.73  20156.68   66.40   66.54   66.27    0.66
300    2400       26332.82  20305.49   67.97   67.70   68.24    0.68
325    2600          18.65  19558.41   65.87   66.67   65.10    0.66
350    2800      128274.11  19918.00   69.19   66.79   71.76    0.69
375    3000        6389.09  19082.18   67.17   64.73   69.80    0.67
400    3200         197.20  18482.39   68.74   68.08   69.41    0.69
425    3400        6237.39  17980.10   69.23   71.55   67.06    0.69
450    3600          50.15  17316.03   69.06   70.33   67.84    0.69
```

Fig. 1. During Training

expansion into other data formats. One area for future scope is implementing this model on PDF datasets, which would involve extracting data from PDFs, annotating it, and converting it into a text format suitable for training the model.

Another potential area for future development is improving the model's ability to handle multilingual resumes. With the increasing globalisation of the job market, applicants often submit resumes in languages other than English. Developing a resume parsing system that can handle multiple languages would be a valuable addition to the current model and could enhance its utility and effectiveness in real-world applications.

VII. Conclusion

In conclusion, the development of a natural language processing system for automated resume parsing and filling job templates can significantly enhance the recruitment process. The proposed system utilizes machine learning techniques such as named entity recognition to extract relevant information from resumes, such as education, work experience, skills, and personal information.

The results of our experiments demonstrate that the proposed system achieves high accuracy and F1 scores for named entity recognition and keyword extraction tasks, indicating its effectiveness in extracting key information from resumes.

However, there are still challenges that need to be addressed in resume parsing, such as handling unstructured and noisy text data, dealing with multilingual resumes, and ensuring data privacy and security.

Despite these challenges, the potential benefits of automated resume parsing are significant, including saving time, reducing errors, and improving the efficiency and effectiveness of the recruitment process. Future research can focus on addressing these challenges and further enhancing the capabilities of automated resume parsing systems to meet the evolving needs of the job market.

References

[1] Bhor, Shubham, Vivek Gupta, Vishak Nair, Harish Shinde, and Manasi S. Kulkarni. ”Resume parser using natural language processing techniques.” Int. J. Res. Eng. Sci 9, no. 6 (2021).

[2] Bhaliya, Nirali, Jay Gandhi, and Dheeraj Kumar Singh. ”NLP based extraction of relevant resume using machine learning.” (2020).

[3] Das, Papiya, Manjusha Pandey, and Siddharth Swarup Rautaray. ”A CV parser model using entity extraction process and big data tools.” IJ Information Technology and Computer Science 9 (2018): 21-31.

[4] Kudatarkar, Vinaya Ramesh, Manjula Ramannavar, and S. S. Nandini. ”A survey on unstructured text analytics approaches for qualitative evalua-tion of resumes.” International Journal of Emerging Technology in Computer Science and Electronics (IJETCSE) April 14 (2015).

[5] Sanyal, Satyaki, Souvik Hazra, Soumyashree Adhikary, and Neelanjan Ghosh. ”Resume parser with natural language processing.” International Journal of Engineering Science 4484 (2017).