Dayananda Sagar University
School of Engineering
Devarakaggalahalli, Harohalli Kanakapura Road, Ramanagara, Karnataka 562112
Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)

Natural language Models

SEMESTER – VI

Course Code: 22AM3610

Resume Parsing Using NER to Auto-Fill Google Forms

Under the guidance of
Prof. Pradeep Kumar K
Prof. Sahil Pocker

Team Member
Name1: Trijal R(ENG22AM0167)
Name2: Pushkar Pallav(ENG22AM0187)
Name3: Utpal Kumar(ENG22AM0139)
Name4: V Ajay(ENG22AM0140)

Contents

Sustainable Development Goal

❖ **Our project supports SDG 8:**

Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all

❖ **Because**

- Our project helps job seekers by streamlining the application process, reducing barriers to employment, and improving efficiency in job searches.
- It supports economic growth by enhancing job accessibility and increasing workforce participation.

SUSTAINABLE DEVELOPMENT GOALS

8 DECENT WORK AND ECONOMIC GROWTH

Introduction

- **Definition:** AI-driven systems that enable machines to understand, process, and generate human language.
- **Technology:** Built using deep learning techniques and trained on vast amounts of text data to recognize patterns, context, and meaning.
- **Popular Models:**
  - **BERT** – Used for text understanding and contextual analysis.
  - **GPT** – Generates human-like text and improves conversational AI.
  - **SpaCy** – Efficient NLP library for tasks like Named Entity Recognition (NER).

Applications of NLMs

- **Chatbots & Virtual Assistants** – Enhancing customer support with AI-powered responses.
- **Sentiment Analysis** – Understanding customer feedback and analyzing social media trends.
- **Machine Translation** – Enabling real-time language translation.
- **Healthcare** – Extracting medical information from clinical notes.
- **Resume Parsing & Recruitment** – Automating candidate screening and shortlisting.

Relevance of the Project

Uses Named Entity Recognition (NER) to extract key details from resumes, including:

Name, Email, Phone number, Education, Experience, Skills

Integration with Google Forms to auto-fill fields, reducing manual data entry.

Key Benefits:

- **Enhances efficiency** by automating resume processing.
- **Reduces errors** compared to manual form-filling.
- **Speeds up recruitment** by minimizing processing time.
- **Improves accuracy** in candidate data extraction.

Literature Survey

Author(s)	Title	Year	Methodology	Key Findings	Limitations
Arvind Kumar Sinha et al.	Automated Resume Parsing and Job Domain Prediction using Machine Learning	2023	NER with ML classifiers (e.g., Decision Tree, SVM)	Achieved 92.08% accuracy in job domain prediction	Model performance may degrade on unstructured or informal resumes
Thangaramya Kalidoss et al.	Automated Resume Parsing and Ranking using Natural Language Processing	2024	NLP for keyword extraction and clustering	Efficient ranking and domain classification of resumes	May not work well for resumes in varied formats or languages
B. V. Brindashree, T. P. Pushpavathi	HR Analytics: Resume Parsing Using NER and Candidate Hiring Prediction	2023	NER + Decision Tree Classifier	Extracted candidate attributes effectively and predicted hiring success	Requires labeled training data and cannot generalize to all industries

Literature Survey

Author(s)	Title	Year	Methodology	Key Findings	Limitations
Lample et al.	Neural Architectures for Named Entity Recognition	2016	BiLSTM-CRF with character-level embeddings	Achieved state-of-the-art results on CoNLL-03	Computationally expensive
Ma and Hovy	End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF	2016	BiLSTM-CNNs-CRF	Improved accuracy with character-level CNNs	High memory requirements
Devlin et al.	BERT: Pre-training of Deep Bidirectional Transformers	2019	BERT fine-tuned for NER tasks	Achieved high accuracy with large datasets	Requires extensive computational resources

Problem Definition

Statement of the Problem

Manually extracting information from resumes and entering it into application forms or databases is a time-consuming and error-prone process. Recruiters and HR professionals often have to handle hundreds or thousands of resumes, making it difficult to efficiently process and organize candidate information. This manual approach slows down hiring and increases the risk of data entry errors.

Gap in Current Solutions

- Many **existing resume parsers** rely on rigid templates and keyword matching, leading to inaccurate extractions when resumes vary in format.
- Most **ATS (Applicant Tracking Systems)** are expensive and may not integrate well with widely used tools like **Google Forms**.
- Traditional **form-filling methods require manual input**, making them inefficient for bulk resume processing.

Why Solving This Problem is Essential

- **Efficiency** – Automating resume parsing saves time and effort, allowing recruiters to focus on candidate assessment.
- **Accuracy** – NER-based extraction ensures structured and error-free data entry.
- **Scalability** – The system can handle large volumes of resumes without additional workload.
- **Cost-Effective** – Unlike expensive ATS solutions, integrating with Google Forms provides a free and widely accessible alternative.

By addressing these challenges, this project aims to **streamline the recruitment process, reduce human effort, and improve hiring speed and accuracy**.

Objectives

The objectives are designed to be **SMART (Specific, Measurable, Achievable, Relevant, and Time-bound)**:

1. **Develop an NER-based system** to accurately extract structured information from resumes (**Specific, Achievable**).
2. **Process and clean resume data** by handling different file formats (PDF, DOCX, TXT) within **four weeks** (**Measurable, Time-bound**).
3. **Identify and classify key entities** (Name, Contact, Education, Experience, Skills) with an extraction accuracy of at least **90%** (**Measurable, Relevant**).
4. **Map extracted information to Google Form fields** using automation tools like Google Forms API or Selenium within **two weeks** (**Achievable, Time-bound**).
5. **Automate the form-filling process** to reduce manual effort by **at least 80%** compared to traditional methods (**Relevant, Measurable**).
6. **Test and evaluate the system** on at least **50 different resumes** to ensure reliability and accuracy before deployment (**Specific, Measurable, Time-bound**).
7. **Enhance system adaptability** by improving entity recognition for resumes with different structures and layouts (**Relevant, Achievable**).

Through this implementation, the project aims to **increase efficiency, minimize human errors, and create a scalable solution for automated resume processing**.

Methodology

1.DataCollection

- Collect a diverse set of resumes in various formats (PDF, DOCX, TXT).
- Ensure data contains personal, educational, and professional details across different domains.

2.Preprocessing

- Convert resumes to plain text using PDF and DOCX parsers.
- Clean and normalize text (remove stopwords, punctuation, special characters).
- Tokenize and segment text into relevant sections (e.g., Education, Skills, Experience).

Named Entity Recognition (NER)

- Use pre-trained NER models (e.g., SpaCy, BERT-NER, or fine-tuned transformers).
- Extract key entities such as:Name, Email ID, Phone Number, Education Qualifications, Skills, Work Experience, Projects, Certifications

4.Field Mapping

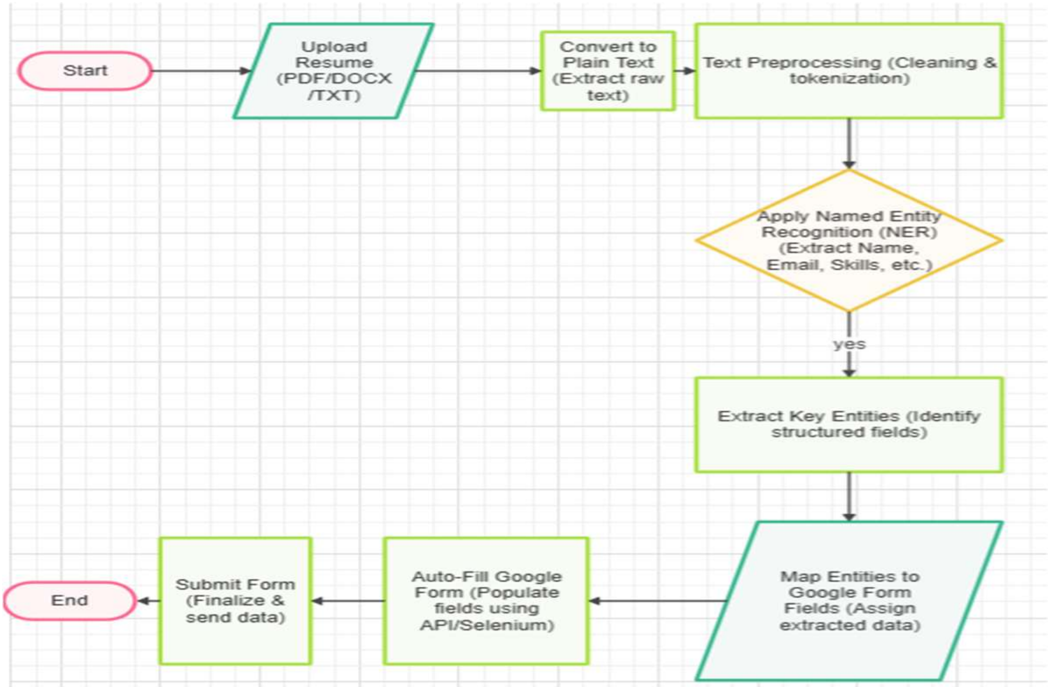
- Map extracted entities to corresponding fields in the Google Form.
- Define a schema that aligns NER tags with form field names.

5.Auto-Fill Automation

- Use Google Forms API or browser automation tools (e.g., Selenium) to auto-fill forms.
- Validate the filled data before submission.

6.Evaluation

- Compare system-generated outputs with manually filled forms to calculate precision, recall, and F1 score of entity extraction.
- Conduct user testing for functional accuracy and time efficiency.



Conclusion

➔ **Efficient and Automated Process** – The developed browser extension successfully streamlines the job application process by automatically filling Google Forms using resume data, reducing manual effort and saving time for users.

➔ **Enhanced Accessibility** – By leveraging NLP, the project improves accessibility for job seekers, making the application process more efficient and user-friendly, especially for those applying to multiple positions.

➔ **Integration of AI and Web Technologies** – The project effectively combines web automation and natural language processing (NLP) to demonstrate the practical applications of AI in the recruitment domain.

➔ **Future Scope and Improvements** – This project can be further enhanced by supporting multiple form formats, improving accuracy in field matching, and integrating with other job application platforms for broader usability.

References

1. **Sinha, A. K., Akhtar, M. A. K., & Kumar, M.** (2023). *Automated Resume Parsing and Job Domain Prediction using Machine Learning*. Indian Journal of Science and Technology, 16(35), 2411–2418. https://doi.org/10.17485/IJST/v16i35.1387

2. **Kalidoss, T., Logeswari, G., Gajendran, S., & Roselind, J. D.** (2024). *Automated Resume Parsing and Ranking using Natural Language Processing*. In *2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT)*. ResearchGate. https://www.researchgate.net/publication/381932058

3. **Brindashree, B. V., & Pushphavathi, T. P.** (2023). *HR Analytics: Resume Parsing Using NER and Candidate Hiring Prediction Using Machine Learning Model*. International Journal of Research in Engineering, Science and Management, 6(7), 276–280. https://journal.ijresm.com/index.php/ijresm/article/view/2900

4. **Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C.** (2016). Neural architectures for named entity recognition. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 260–270.

5. **Ma, X., & Hovy, E.** (2016). End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, 1064–1074.

6. **Devlin, J., Chang, M. W., Lee, K., & Toutanova, K.** (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 4171–4186.

Acknowledgments

We would like to express our sincere gratitude to **Prof. Pradeep Kumar K** and **Prof. Sahil Pocker** for providing us with the opportunity to work on this project. Their invaluable guidance, encouragement, and insights have been instrumental in shaping our understanding and execution of this work.

We would also like to acknowledge the use of various online resources, including the **WEB**, which provided us with essential information, and **ChatGPT**, which assisted in refining our ideas and enhancing our implementation.

Lastly, we appreciate the collaborative efforts of our team members, whose dedication and contributions made this project possible.