IJARSCT

ISSN (Online) 2581-9429

IJARSCT

Impact Factor: 7.301

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Volume 3, Issue 1, April 2023

Resume Parser and Summarizer

Er. Farzana Khan[1], Hamdan Patel[2], Arshad Shaikh[3], Fawzah Sayed[4], Abdul Rehman Soorya[5]

Assistant Professor, Department of Information Technology[1]

Student, Department of Information Technology[2,3,4,5]

M. H. Saboo Siddik College of Engineering, Mumbai, Maharashtra, India

Abstract: *Resume parsing is a highly important process for HR departments and recruiters looking to streamline their hiring process. By converting a resume into structured data, it allows for easy organization and quick searchability for specific qualifications and skills. However, there are still limitations to the technology, as language can be complex and ambiguous. It is important for companies to stay up to date with advancements in Natural Language Processing and Artificial Intelligence to improve the accuracy of resume parsing and avoid overlooking qualified candidates. Overall, resume parsing is a crucial tool for modern recruitment, but it still requires human oversight to ensure the best candidates are not overlooked.*

Keywords: Artificial Intelligence, Machine Learning, Natural Language Processing, Resume, Parser.

I. INTRODUCTION

A resume parser is a deep learning/AI framework that extracts complete information from resumes, analyses it, stores it, organizes it, and enriches it using taxonomies. Resume parsing software expedites and improves the hiring process. Quick and precise resume parsing technology increases efficiency and provides a better candidate experience. A resume parser is an interpreter or compiler that converts unstructured data into structured data. It is a component that automatically categorizes information such as contact information, educational qualifications, work experience, skills, achievements, and professional certifications into various fields and parameters to assist you in quickly identifying the most relevant resumes based on your criteria. Resume parsers have achieved up to 87% accuracy, which refers to data entry accuracy and correctly categorizing data. Because human accuracy is typically less than 96%, the resume parsers achieved "near human accuracy”. To compare data entry accuracy, one executive recruiting firm tested three resume parsers and humans. They ran 1000 resumes through resume parsing software before manually parsing and entering the data. The company hired a third party to assess how the humans performed in comparison to the software. They discovered that the resume parser results were more comprehensive and contained fewer errors. Humans did not enter all of the information on the resumes and occasionally misspelt words or spelt numbers incorrectly. A resume for an ideal candidate was created based on the job description for a clinical scientist position in a 2012 experiment. Due to the date being listed before the employer, one of the candidate's work experiences was completely lost after going through the parser. Several educational degrees were also missed by the parser. As a result, the candidate received a relevance ranking of 43%. If this had been a real candidate's resume, they would not have advanced to the next stage despite being qualified for the position.

A similar study on current resume parsers to see if there have been any improvements over the last few years would be beneficial. Marianne Bertrand and Sendhil Mullainathan conducted a famous study in 2003 to see if candidates with the names Emily and Greg were more employable than Lakisha and Jamal. Resumes with white-sounding names received 50% more call-back’s than resumes with black-sounding names[14]. In 2014, a study was conducted in Australia and New Zealand to investigate gender-based name discrimination. Insync Surveys, a research firm, and Hays, a recruitment specialist, each sent a resume to 1,029 hiring managers with the only difference being the name. Half of the hiring managers received Simon Cook's resume, while the other half received Susan Campbell's resume. According to the study, Simon was more likely to get a call-back.

DOI: 10.48175/IJARSCT-9064

ISSN 2581-9429 IJARSCT
INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN SCIENCE, COMMUNICATION AND TECHNOLOGY

II. EXISTING SYSTEM

- Affinda: Affinda is based on a distinct set of AI technology models. These avoid the traditional rule-driven approach infavour of machine learning and probabilistic predictive methods for more accurate results. This allows the Affinda CV parser to extract information from your dataset more accurately [11].
- Zoho Recruit: Zoho Recruit is a cloud-based applicant tracking system. Zoho provides a full suite of applicant tracking system features, such as social recruiting and resume parsing. Its custom automatable reports enable processing analysis and information sharing. It is available in three flavours': free, standard, and enterprise [12].
- hireEZ: hireEZ's outbound recruitment software aims to transform today's hiring technology in order to build the workforce of the future. Intelligent sourcing, engagement, analysis, and integrations help organizations scale while working with existing platforms. Customers use hireEZ to source across 45+ platforms and 750M+ open web professional profiles, quickly gather contact information, and hire efficiently with team collaboration, pipeline management, and intelligent engagement, according to the vendor [13].

III. PROPOSED SYSTEM

The system will assist recruiters in viewing the summarized resumes and hiring the best candidate. The resumes dataset was gathered from various websites such as Kaggle and GitHub. The dataset is used to train the model. Our website's home page includes a login and sign in option. The login and sign in options will be for two different people, namely the candidate and the recruiting company. The candidate will upload his or her resume, which will be saved in a database. The output screen will display summarized data from the database.

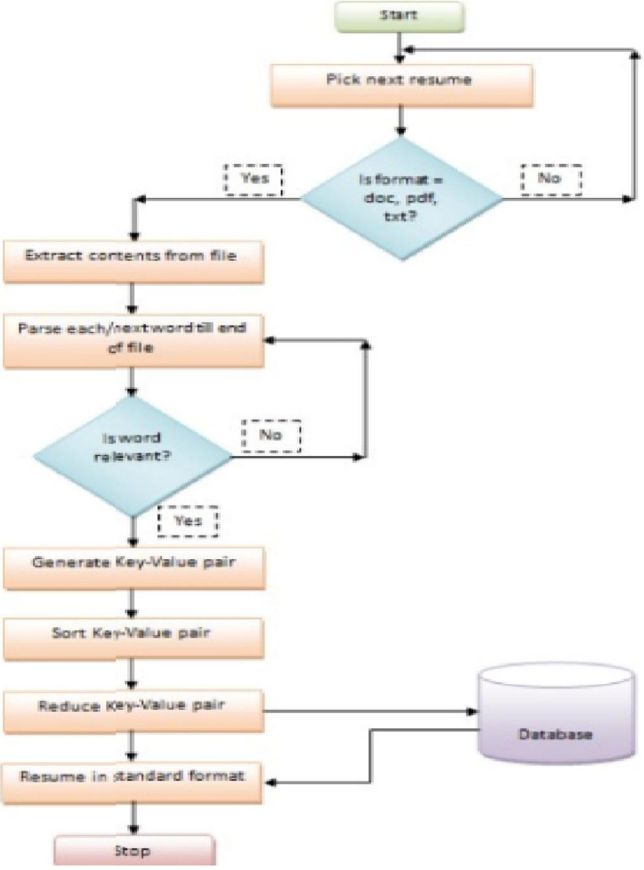


Fig 1 Flowchart of the Proposed System

For job candidates, a GUI-based webpage will appear where they can upload their resume in any format. When they upload their resumes, they will be saved in our database as a standard. They will be stored in our database as a standard readable structured data format once they upload their resumes. Following that, the recruiters will use a GUI-based website to organize the information they require based on their preferences. Candidate points will be computed based on recruiters' needs, and candidate rankings will be generated based on that, with recruiters receiving a list of candidate rankings based on their preferences. Our project will be divided into several modules, such as:

- Creating a Domain: Because the proposed system is domain independent and will be used by a large number of users, this module is responsible for creating user accounts and databases.
- Registration or Login Module: If a new user wishes to interact with our system, he must first register by providing all of the necessary information (validation). If the user already exists, he must log in first.
- Parsing and Ranking: The parsing module is in charge of processing the document and saving it in json format for later use by the ranking module. The ranking module will then use the json file to rate the information of the candidates based on their abilities, and the data will be saved in the database

Our system will detect the resume format (pdf, text, docx, rift, html) and extract the content from the files. The system will then parse each extracted piece of content and check it. Our system will detect the resume format (pdf, text, docx, rift, html) and extract the content from the files. The system will then parse each extracted piece of content and check it. Our system will detect the resume format (pdf, text, docx, rift, html) and extract the content from the files. The system will then parse each extracted content and determine whether it is relevant for generating key value pairs. If the content extracted from the file is not relevant, it will re-examine the file for relevant data. Sorting of key value pairs occurs from the extracted key value pairs, and the important key value pairs are extracted from the sorted key value pairs. All key-value pairs are saved in the database. The system will extract the summarized and standardized resume from the database and provide the output.The datasets for the project were gathered from various websites such as Kaggle and GitHub. The resumes were in various formats and had varying educational backgrounds [1],[2],[3],[4],[5].

IV. TOOLS AND TECHNOLOGIES USED

1. Python: Python is a general-purpose, high-level programming language. Its design philosophy prioritizes code readability by employing significant indentation. Python is garbage-collected and dynamically typed. It supports a wide range of programming paradigms, including structured (especially procedural), object-oriented, and functional programming. Because of its extensive standard library, it is frequently referred to as a "batteries included" language. Guido van Rossum began developing Python as a successor to the ABC programming language in the late 1980s, and it was first released in 1991 as Python 0.9.0. Python 2.0 was released in the year 2000. Python 3.0, released in 2008, was a significant revision that was not fully backward-compatible with previous versions. Python 2.7.18, released in 2020, was the final Python 2 release. Python is consistently ranked among the top programming languages[6][7].
2. MongoDB: MongoDB is a cross-platform document-oriented database programme that is open source. MongoDB, a NoSQL database programme, employs JSON-like documents with optional schemas. MongoDB was created by MongoDB Inc. and is licensed under the Server-Side Public License (SSPL), which several distributions consider to be non-free. MongoDB is a MACH Alliance member.
3. HTML: HTML, or Hypertext Markup Language, is the standard markup language for documents intended to be displayed in a web browser. It is frequently aided by technologies like Cascading Style Sheets (CSS) and scripting languages like JavaScript. Web browsers receive HTML documents from a web server or local storage and convert them to multimedia web pages. HTML semantically describes the structure of a web page and originally included visual cues.
4. CSS: Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.
5. Flask: Flask is a Python-based micro web framework. It is classified as a microframework because it does not necessitate the use of any specific tools or libraries. It lacks a database abstraction layer, form validation, and other components where third-party libraries provide common functions. Flask, on the other hand, supports extensions that can add application features as if they were built into Flask itself. There are extensions for

DOI: 10.48175/IJARSCT-9064

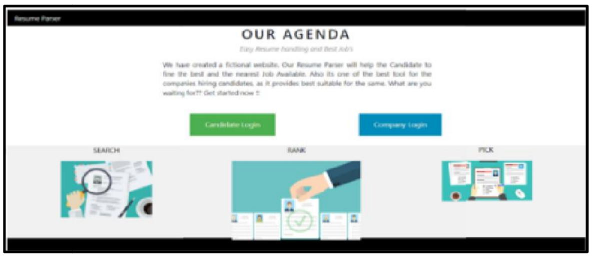
ISSN
2581-9429
IJARSCT

object-relational mappers, form validation, upload handling, various open authentication technologies, and several framework-related tools [8].

6. Natural Language Processing: Natural language processing (NLP) is an interdisciplinary subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data. The goal is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights contained in the documents as well as categorize and organize the documents themselves. Challenges in natural language processing frequently involve speech recognition, natural-language understanding, and natural-language generation [9].
7. SpaCy: SpaCy is an open-source software library for advanced natural language processing, written in the programming languages Python and Cython. The library is published under the MIT license and its main developers are Matthew Honnibal and Ines Montani, the founders of the software company Explosion [10].

Fig 2 SpaCy

V. IMPLEMENTATION SCREEN

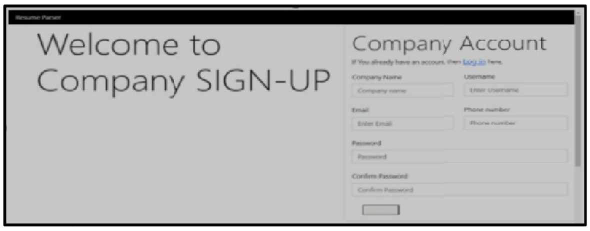


OUR AGENDA

Easy Resume handling and Best Job's

We have created a fictional website. Our Resume Parser will help the Candidate to fine the best and the nearest Job Available. Also its one of the best tool for the companies hiring candidates, as it provides best suitable for the same. What are you waiting for?? Get started now !!

Fig 3 Home-page



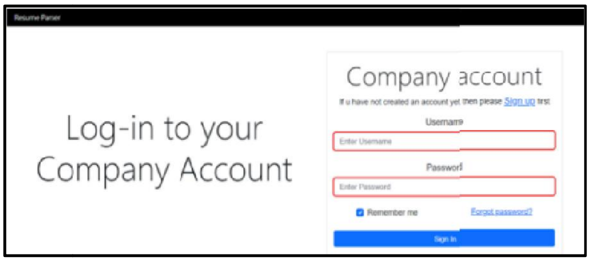
Welcome to Company SIGN-UP

Company Account

If You already have an account, then Log in here.

Company Name: Company name
Username: Enter Username
Email: Enter Email
Phone number: Phone number
Password: Password
Confirm Password: Confirm Password

Fig 4 Sign-Up Page



Log-in to your Company Account

Company account

If u have not created an account yet then please Sign up first

Username: Enter Username
Password: Enter Password

☑ Remember me
Forgot password?

Fig 5 Login Page

VI. CONCLUSION

People nowadays value their time and the ease with which they can complete their tasks. A resume parser is an interpreter or compiler that converts unstructured data into structured data. It is a component that automatically categorizes information such as contact information, educational qualifications, work experience, skills, achievements, and professional certifications into various fields and parameters to assist you in quickly identifying the most relevant resumes based on your criteria. The resume parser will assist the recruiting company in quickly and easily parsing and summarizing resumes. It will help the recruiting firm parse multiple resumes at the same time. The resume parser will support a variety of document types, including docx, pdf, and html. The resume parser will help various recruiting firms find candidates with the necessary experience and competencies. A resume parser will improve the recruitment process's efficiency and eliminate human errors

VII. FUTURE SCOPE

For candidate resumes, we can provide a ranking. Security features such as checking resumes for viruses and rejecting those that contain them can be included. The website can be expanded to include a variety of skill training courses. An open service for all candidates can be added to the website that will rank the candidates based on their skills and save that information in a database. Later, the same resumes will be used by some recruiting companies if they want to hire a specific candidate for a specific job profile. This website can be linked to job search websites.

VIII. ACKNOWLEDGMENT

We would like to express our sincere gratitude to all the individuals and organizations who have supported and contributed to the successful completion of this project. First and foremost, we would like to thank our project supervisor, Er. Farzana Shaikh, and our in-charge Head of Department, Er Shrinidhi Gindi and Faculty of Information Technology, Er. Sanam Kazi, for their invaluable guidance, insights, and unwavering support throughout the project. Their expertise and encouragement have been instrumental in shaping our ideas and helping us to achieve our goals. We would also like to extend our appreciation to all the participants who volunteered their time and effort to provide us with the necessary data and information for the project. Without their cooperation, this project would not have been possible. We would also like to express our heartfelt appreciation and gratitude to our in-charge Principal Dr. Ganesh Kame for his assistance and facilities in this regard. Finally, we'd like to thank everyone who contributed directly or indirectly to the project's success.

Once again, we express our heartfelt thanks to everyone who has contributed to this project's success

REFERENCES

[1]. Sankar, A. (2013). “Towards an automated system for intelligent screening of candidates for recruitment using ontology mapping (EXPERT)”. International Journal of Metadata, Semantics and Ontologies, 8(1), 56. https://doi.org/10.1504/ijmso.2013.054184. (Accessed 25 March 2023)

[2]. Jagan Mohan Reddy D, Sirisha Regella., “Recruitment Prediction using Machine Learning”, IEEE Xplore, 2020. (Accessed 25 March 2023)

[3]. Färber,F., Weitzel, T.,Keim, T., 2003. “An automated recommendation approach to selection in personnel recruitment”. AMCIS 2003 proceedings, 302. (Accessed 25 March 2023)

[4]. Chirag Daryania, Gurneet Singh Chhabrab, Harsh Patel, Indrajeet Kaur Chhabrad, Ruchi Patel., “An Automated Resume Screening System using Natural Language Processing and Similarity”. (2020). Topics In Intelligent Computing and Industry Design. (Accessed 26 March 2023)

[5]. Momin Adnan, Gunduka Rakesh, Juneja Afza, Rakesh Narsayya Godavari, Gunduka and Zainul Abideen Mohd Sadiq Naseem., “Resume Ranking using NLP and Machine Learning”, (2016b). Institutional Repository of the Anjuman-I-Islam’s Kalsekar Technical Campus. (Accessed 25 March 2023)

[6]. Zdena Dobesova, “Programming language Python for data processing”, (2011), 2011 International Conference on Electrical and Control Engineering. DOI 10.1109/ICECENG.2011.6057428. (Accessed 25 March 2023)

[7]. Abhinav Nagpal and Goldie Gabrani, “Python for Data Analytics, Scientific and Technical Applications”, (2019), 2019 Amity International Conference on Artificial Intelligence (AICAI). DOI 10.1109/AICAI.2019.8701341. (Accessed 25 March 2023)

DOI: 10.48175/IJARSCT-9064

ISSN
2581-9429
IJARSCT

[8]. Patrick Vogel, Thijs Klooster, Vasilios Andrikopoulos and Mircea Lungu, “A Low-Effort Analytics Platform for Visualizing Evolving Flask-Based Python Web Services”, (2017), 2017 IEEE Working Conference on Software Visualization (VISSOFT), DOI: 10.1109/VISSOFT.2017.13. (Accessed 26 March 2023)

[9]. P. Jacobs and T. Patten, “Natural-language processing”, (1994), IEEE Expert (Volume: 9, Issue: 1, February 1994), DOI: 10.1109/64.295134. (Accessed 26 March 2023)

[10]. Vivek Anand, Bhupendra Singh Tyagi, Ashish Kumar and Swaranjali Jugran, (2021), “Extractive Automatic Text Summarization using SpaCy in Python & NLP”, 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), DOI: 10.1109/ICACITE51222.2021.9404712. (Accessed 26 March 2023)

[11]. Affinda, 27 March 2023, https://www.affinda.com/

[12]. Zoho Recruit, 27 March 2023, https://www.zoho.com/recruit/

[13]. HireEZ, 27 March 2023, https://hireez.com/

[14]. American Economic Association, https://www.aeaweb.org/articles?id=10.1257/0002828042002561, 27 March 2023.