"CareerBoostAI: Resume and Github Optimization Platform"

B. KUMARi, DHARMIREDDY MANIKANTA

Assistant professor, MCA Final Semester, Master of Computer Applications, Sanketika Vidya Parishad Engineering College, Vishakhapatnam,
Andhra Pradesh, India.

ABSTRACT:

The Advanced ATS Resume Checker is a web-based, AI-powered tool designed to help job seekers optimize their resumes for Applicant Tracking Systems (ATS) and increase job application success rates. Built using Python and Streamlit, it offers multi-faceted resume analysis including keyword matching, formatting validation, and industry-specific optimization. Additionally, it integrates GitHub profile analysis to cross-reference technical skills and provides personalized, AI-driven recommendations. The platform also delivers industry-specific insights such as top required skills and formatting best practices. Users can interact with an intuitive UI to analyze resumes, compare job descriptions, and improve their career documents through visual feedback tools like heatmaps and word clouds.

Index terms: Resume Analysis, Keyword Matching, ATS Optimization, Resume Formatting Check, Resume Heatmap, Word Cloud, Industry-Specific Advice, GitHub Profile Analyzer, Repository Analysis, Language Distribution, AI-Powered Insights, Technical Skill Validation.

1. INTRODUCTION:

In today's competitive job market, resumes often face their first screening by Applicant Tracking Systems (ATS)—automated tools that filter out applications based on keyword relevance, formatting, and content structure. Unfortunately, many qualified candidates are rejected before a human ever sees their resume due to poor ATS compatibility.The Advanced ATS Resume Checker is an AI-powered web application designed to bridge that gap. It empowers job seekers to optimize their resumes using intelligent analysis, keyword matching, formatting validation, and industry-specific insights. The tool provides a comprehensive evaluation of resumes against job descriptions and helps highlight strengths, identify weaknesses, and deliver actionable recommendations.An additional feature of the platform is the GitHub Profile Analyzer, which offers technical candidates a way to assess how well their coding portfolio supports their resume. This tool provides insights into repository relevance, programming language distribution, contribution activity, and project visibility (stars, forks). It also detects skill gaps between the resume and GitHub activity, thereby giving users a complete picture of their technical persona.Built with Python, Streamlit, NLTK, Pandas, Matplotlib, and Plotly, and integrated with Google's Generative AI, the application provides a seamless and interactive user experience. It is designed for job seekers, developers, career coaches, and hiring professionals who wish to gain a data-backed understanding of a candidate's resume and technical profile.

1.1 EXISTING SYSTEM:

Existing resume analysis tools like Jobscan, Zety, and Resumeworded primarily rely on keyword matching and formatting suggestions to help users improve their resumes. These systems scan resumes for common terms in a job description and provide a match score based on keyword frequency. While helpful, they are often limited by rule-based logic and lack contextual understanding of candidate experience. Most tools do not evaluate the semantic relevance of skills or accomplishments, which reduces the depth and accuracy of their recommendations.Additionally, existing systems do not integrate technical profile analysis, such as GitHub data, which is crucial for evaluating software engineers and developers. They miss out on assessing real-world coding experience, project involvement, and open-source contributions. Moreover, many platforms offer limited visual feedback, making it hard for users to interpret results. These limitations highlight the need for a more intelligent, ML-powered system like the Advanced ATS Resume Checker, which combines resume parsing, job description matching, and GitHub analysis into one unified platform.

1.1.1 CHALLENGES:

1. Unstructured Resume Formats: Difficulty in extracting clean and consistent text from varied PDF layouts (tables, columns, graphics.

2. Semantic Matching Limitations: Traditional models like TF-IDF couldn't fully understand the context or meaning of job requirements vs. resume content.

3. Lack of Labeled Training Data: Absence of large, real-world datasets made supervised learning approaches difficult to implemen.

4. Performance vs. Accuracy Trade-off: Ensuring real-time feedback while maintaining the quality of machine learning analysis required model optimization,

5. GitHub Data Inconsistencies: Challenges in filtering meaningful repositories and normalizing diverse contribution patterns for technical evaluation.

6. User Interpretability: Translating complex ML results into clear, actionable insights that are easy for non-technical users to understand.

1.2 PROPOSED SYSTEM:

The proposed system, Advanced ATS Resume Checker, aims to overcome the limitations of existing resume analysis platforms by integrating Machine Learning (ML) and Natural Language Processing (NLP) for intelligent, context-aware resume evaluation. Unlike traditional keyword matchers, this system performs semantic analysis to understand the actual meaning behind words and phrases, offering more accurate job-resume alignment. The tool provides multiple modes of analysis, including quick scan, ATS optimization, formatting checks, and detailed reports—making it suitable for users across various experience levels and industries.

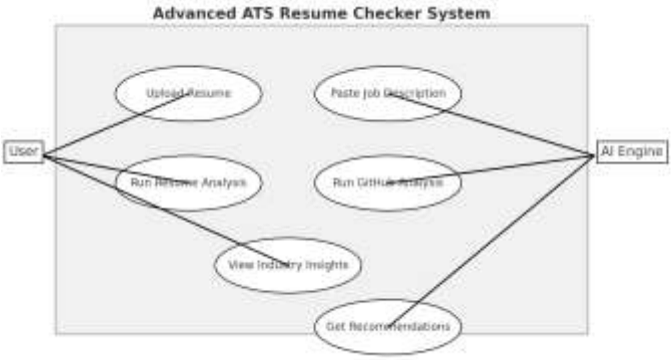


fig 1 system architecture of proposed system

1.2.1 ADVANTAGES:

1. **Semantic Matching:** Uses NLP and ML to understand context, not just keywords, for more accurate job-resume alignment.

2. **Hybrid CNN-LSTM Architecture:**Combines CNN (for spatial feature extraction) and LSTM (for capturing temporal dependencies), enhancing diagnostic precision and contextual understanding of lung sounds.

3. **Visual Feedback:** Offers resume heatmaps, word clouds, and skill gap charts to help users easily understand their strengths and areas for improvement

4. **Multiclass Disease Detection:**Unlike many existing models focused on binary classification, this system classifies seven different pulmonary diseases, providing richer diagnostic detail.

5. **Multiple Analysis Modes:** Supports Quick Scan, ATS Optimization, Detailed Analysis, and Formatting Check, giving users flexibility.

2. LITERATURE REVIEW:

In recent years, the use of Applicant Tracking Systems (ATS) has become widespread across industries, especially in large-scale recruitment. These systems automate the initial screening of resumes by scanning for relevant keywords, formatting structures, and matching applicant profiles to job descriptions. Several studies highlight that over 70% of resumes are rejected by ATS before reaching a recruiter, emphasizing the need for job seekers to optimize their resumes for these systems.Parallel to this, the increasing demand for technical roles has led to a surge in candidates maintaining online portfolios and GitHub repositories. Research shows that technical hiring managers often evaluate candidates based on real-world coding experience, contributions to open-source projects, and practical skill demonstrations. However, most resume optimization tools do not incorporate GitHub analysis, missing a key dimension of a candidate's technical capabilities.

2.1 ARCHITECTURE:

The architecture of the **Advanced ATS Resume Checker** follows a modular, AI-driven, web-based design that ensures scalability, accuracy, and user-friendly interaction. The **frontend**, built with Streamlit, provides an intuitive interface where users can upload resumes, paste job descriptions, and view results through visual tools such as heatmaps, word clouds, and keyword match charts. The **backend**, developed in Python, processes the input through dedicated modules including a resume parser, keyword matching engine, formatting validator, industry insights generator, and AI-powered recommendation engine. These modules leverage NLP techniques for skill extraction, keyword relevance scoring, and formatting compliance checks. An **integration layer** connects with the GitHub API to analyze public repositories, cross-referencing coding activity with the technical skills listed in the resume

Input Module – Data Acquisition:

The Input Module – Data Acquisition in the Advanced ATS Resume Checker is responsible for collecting and preparing all necessary input data for analysis. It serves as the entry point of the system, enabling users to upload resumes in formats such as PDF or DOCX, paste or upload job descriptions, and optionally provide a GitHub username for profile analysis. This module uses file parsing tools like pdfplumber and docx2txt to extract raw text, structure, and metadata from the uploaded documents while maintaining section integrity (e.g., education, work experience, skills). For job descriptions, it captures role-specific keywords, qualifications, and responsibilities to create a benchmark for ATS scoring. In the case of GitHub integration, the module connects to the GitHub API to fetch repository data, commit history, and language statistics. All acquired data is sanitized, validated, and stored temporarily in memory to ensure security and privacy before being passed to downstream processing modules for keyword matching, formatting checks, and AI-driven recommendations

ER Diagram:

The provided ER (Entity-Relationship) Diagram represents a structured model comprising multiple entities, their attributes, and the relationships between them. Each entity box contains a list of attributes that describe its properties, such as Gapd, Ouper, Orepeen, and others. The diagram shows how these entities are interconnected through relationships labeled A and B, which are depicted using diamond shapes. These relationships are further clarified by terms like Redinality, Readinality, Cardinality, and Suacity, indicating the nature and constraints of these connections.

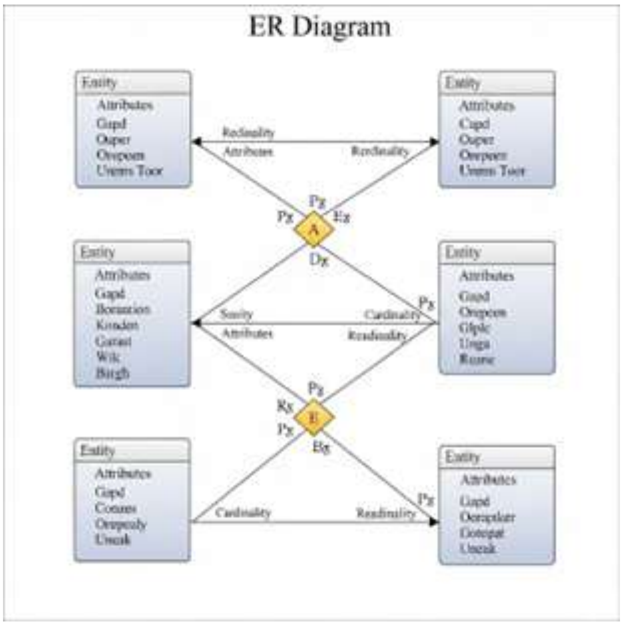


fig 2 system architecture of ER Diagram

Activity Diagram:

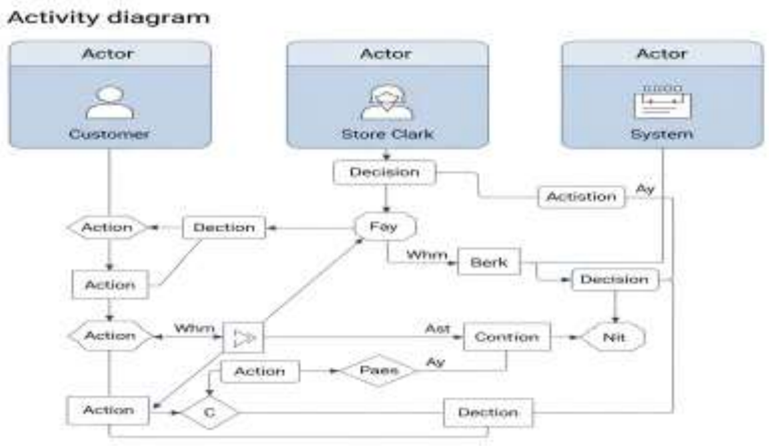


Fig3 Activity diagram

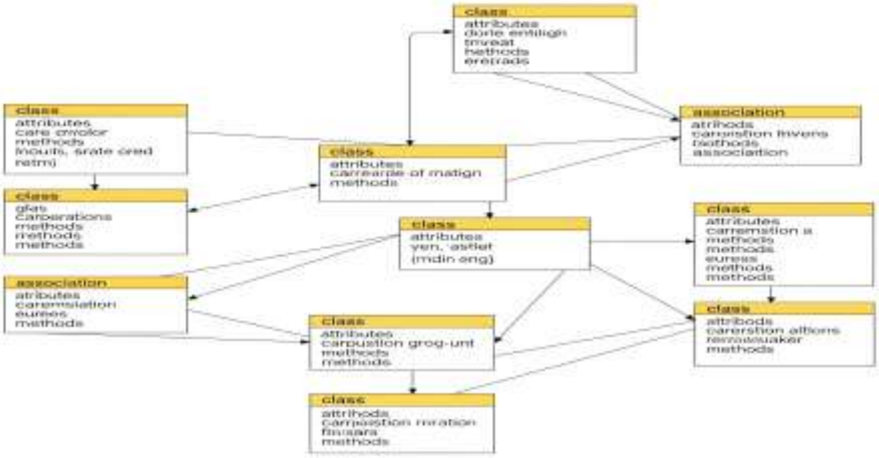


Fig4 class diagram

2.2 ALGORITHM

TF-IDF is a statistical technique used to determine how important a word is in a document relative to a collection of documents (called a corpus). It increases with the number of times a word appears in a document but is offset by how often the word appears in other documents. In the context of resume analysis, TF-IDF helps identify the most relevant keywords in both the resume and the job description. Common words like "and" or "the" are given low weight, while unique terms like "Python" or "cloud computing" are given higher importance. This creates a vector representation of the document based on keyword relevance. Each document (resume and job description) becomes a numerical vector..Cosine similarity is a mathematical formula that calculates the angle between two vectors in a multi-dimensional space. When applied to TF-IDF vectors of a resume and a job description, it returns a similarity score between 0 and 1. A score closer to 1 means the documents are highly similar in terms of content and keyword usage. This is particularly useful in ATS systems to match resumes with relevant job descriptions. Unlike raw word counting, cosine similarity accounts for both term frequency and overall context. It ensures that even if exact words differ slightly, relevance can still be detected.. These layers are pivotal in combining the learned features and performing the final classification or regression tasks. The output layer, often using a softmax activation function, generates a probability distribution over the output classes in classification tasks.

2.3 TECHNIQUES

The **Advanced ATS Resume Checker** utilizes a blend of **Natural Language Processing (NLP)**, **data visualization**, and **API-driven integrations** to deliver accurate, ATS-friendly resume optimization. NLP techniques such as tokenization, lemmatization, stemming, and stopword removal are applied to normalize text and focus on meaningful terms, while Named Entity Recognition (NER) identifies skills, tools, certifications, and qualifications. Relevance between resume content and job descriptions is quantified using cosine similarity and TF-IDF scoring, enabling precise keyword matching and scoring that considers frequency, placement, and context. Formatting validation techniques analyze document structure to ensure ATS compatibility, detecting potential parsing issues like inconsistent fonts, tables, or embedded images. GitHub API integration allows for repository analysis, commit tracking, and language distribution mapping, which are cross-referenced with resume-listed skills for authenticity verification. Data visualization methods such as heatmaps, word clouds, and interactive charts present keyword density, skill match percentages, and ATS score breakdowns in a clear, engaging format..

2.4 TOOLS

The **Advanced ATS Resume Checker** is built using a comprehensive set of tools that ensure accurate analysis, intuitive visualization, and secure processing. The backend is developed in **Python**, leveraging **Streamlit** for an interactive, web-based frontend. For Natural Language Processing (NLP), it uses **spaCy** for Named Entity Recognition (NER) and tokenization, **NLTK** for stemming, lemmatization, and stopword removal, and **scikit-learn** for TF-IDF vectorization, cosine similarity scoring, and keyword matching. Resume text extraction is handled by **pdfplumber** for PDF files and **docx2txt** for DOCX formats. Data visualization is powered by **Matplotlib** and **Plotly** for charts and section analysis, along with **WordCloud** for highlighting keyword frequency. Integration with the **GitHub API** enables repository analysis, commit tracking, and language distribution mapping. Data processing and manipulation are managed using **Pandas**, with all files stored temporarily in memory to protect user privacy. Security is reinforced through input sanitization and API authentication, ensuring both **fast performance** and **safe handling of sensitive data**.

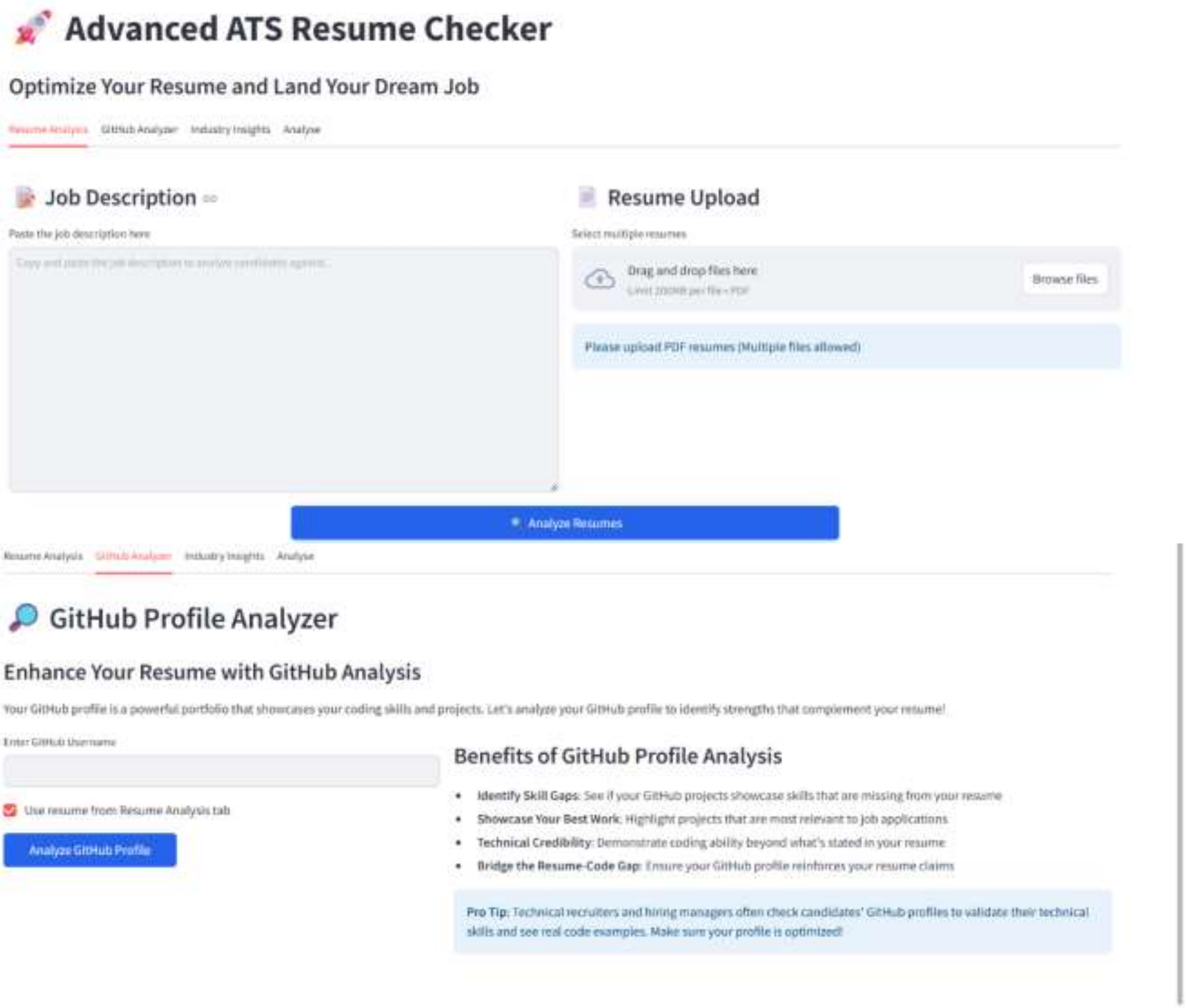
Software Requirement:

Software requirements outline the specific software components, tools, and systems needed to support the development, deployment, and operation of a software system. Here's a detailed breakdown of typical software requirements for a system involving data processing, machine learning, and model inference.

- Domain: Python
- Version: Python IDLE (3.8.10)
- Code Editors: PyCharm, Vscode
- Frameworks and Dependencies: numpy, Streamlit, pandas, Cv2
- Operating System: Windows 11

2.5 METHODS:

The **Advanced ATS Resume Checker** follows a structured, multi-stage methodology to ensure comprehensive analysis and optimization of resumes for ATS compatibility. The process begins with **data acquisition**, where users upload resumes in PDF or DOCX formats, provide job descriptions, and optionally submit a GitHub username for technical skill verification. The **resume parsing** stage extracts text, section headers, and formatting details using tools like pdfplumber and docx2txt, ensuring the structural integrity of key sections such as skills, experience, and education. Next, **Natural Language Processing (NLP)** techniques—such as tokenization, lemmatization, stopword removal, and Named Entity Recognition (NER)—are applied to both the resume and job description to identify and normalize skills, tools, and qualifications. The **keyword matching engine** then calculates ATS relevance scores using methods like TF-IDF vectorization and cosine similarity, assigning weighted scores based on keyword frequency, placement, and contextual relevance



Advanced ATS Resume Checker

Optimize Your Resume and Land Your Dream Job

Job Description

Paste the job description here

Copy and paste the job description to analyze candidates against...

Resume Upload

Select multiple resumes

Drag and drop files here

Limit 200MB per file • PDF

Please upload PDF resumes (Multiple files allowed)

GitHub Profile Analyzer

Enhance Your Resume with GitHub Analysis

Your GitHub profile is a powerful portfolio that showcases your coding skills and projects. Let's analyze your GitHub profile to identify strengths that complement your resume!

Enter GitHub Username

☑ Use resume from Resume Analysis tab

Benefits of GitHub Profile Analysis

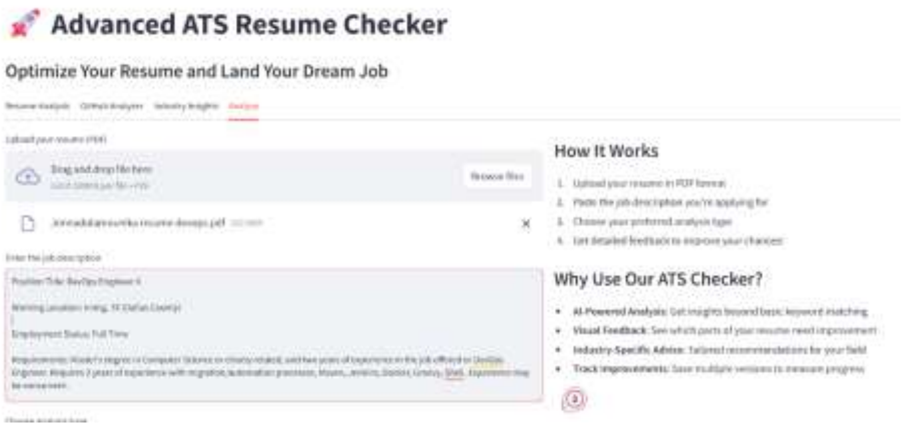
- **Identify Skill Gaps**: See if your GitHub projects showcase skills that are missing from your resume
- **Showcase Your Best Work**: Highlight projects that are most relevant to job applications
- **Technical Credibility**: Demonstrate coding ability beyond what's stated in your resume
- **Bridge the Resume-Code Gap**: Ensure your GitHub profile reinforces your resume claims

Pro Tip: Technical recruiters and hiring managers often check candidates' GitHub profiles to validate their technical skills and see real code examples. Make sure your profile is optimized!

3. METHODOLOGY:

3.1 INPUT:

The input phase of the **Advanced ATS Resume Checker** focuses on acquiring all necessary data required for comprehensive resume analysis. Users interact with the system through a **Streamlit-based web interface**, where they can upload resumes in PDF or DOCX formats, paste or upload job descriptions, and optionally enter a GitHub username for technical skill validation. Upon file upload, the system employs **document parsing tools** such as pdfplumber and docx2txt to extract raw text, maintain section structure, and capture essential metadata without altering the original content.



Advanced ATS Resume Checker

Optimize Your Resume and Land Your Dream Job

How It Works

1. Upload your resume in PDF format
2. Paste the job description you're applying for
3. Choose your preferred analysis type
4. Get detailed feedback to improve your chances

Why Use Our ATS Checker?

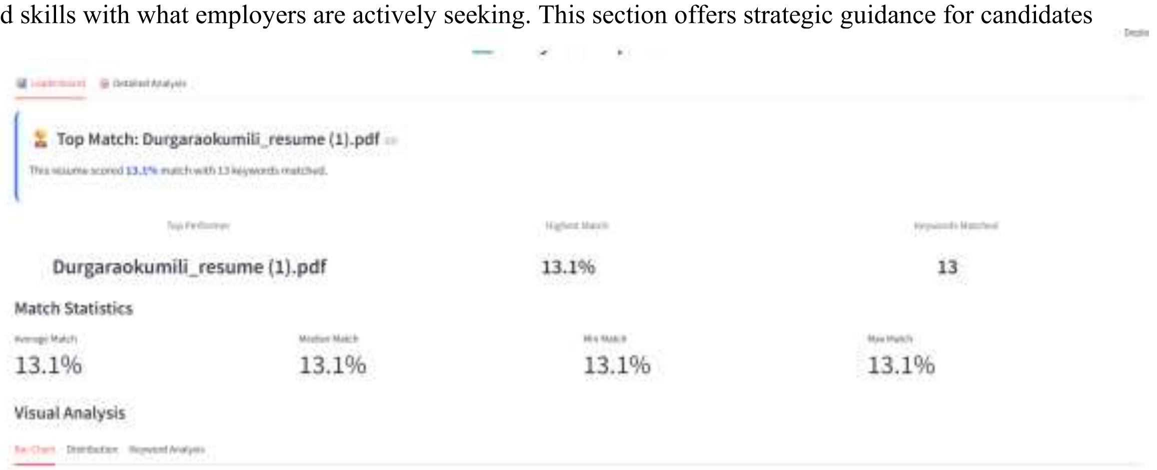
- **AI-Powered Analysis:** Get insights beyond basic keyword matching
- **Visual Feedback:** See which parts of your resume need improvement
- **Industry-Specific Advice:** Tailored recommendations for your field
- **Track Improvements:** Save multiple versions to measure progress

3.2 METHOD OF PROCESS:

The processing stage of the **Advanced ATS Resume Checker** follows a systematic, multi-layered approach to transform raw input data into actionable ATS optimization insights. Once resumes, job descriptions, and optional GitHub profiles are acquired, the system begins with **resume parsing**, using tools like pdfplumber and docx2txt to extract clean text, maintain section structure, and identify formatting elements such as headings, bullet points, and fonts. Next, the extracted content undergoes **Natural Language Processing (NLP)**, where tokenization, lemmatization, and stopword removal prepare the text for analysis. **Named Entity Recognition (NER)** is applied to detect skills, tools, certifications, education, and experience, while **TF-IDF vectorization** and **cosine similarity scoring** compare the resume content against the job description to measure keyword relevance, The **keyword matching engine** assigns weighted scores based on keyword frequency, placement within strategic sections, and contextual relevance. Simultaneously, the **formatting validation module** checks for ATS-friendly structures and flags issues like inconsistent fonts, use of tables, special characters, or non-standard layouts that could hinder ATS parsing

3.3 OUTPUT:

The dashboard of the Advanced ATS Resume Checker provides a clean and intuitive interface for users to analyze and optimize their resumes. The design is divided into two primary sections: Job Description Input and Resume Upload. The GitHub Analyzer tab offers users the ability to enhance their resume analysis by evaluating their GitHub profile. This module bridges the gap between resume content and real-world coding contributions, offering technical recruiters a transparent view of the candidate's programming capabilities, The Industry Insights tab equips job seekers with up-to-date data and trends from their desired field, helping them align their resume and skills with what employers are actively seeking. This section offers strategic guidance for candidates



Top Match: Durgaraokumili_resume (1).pdf

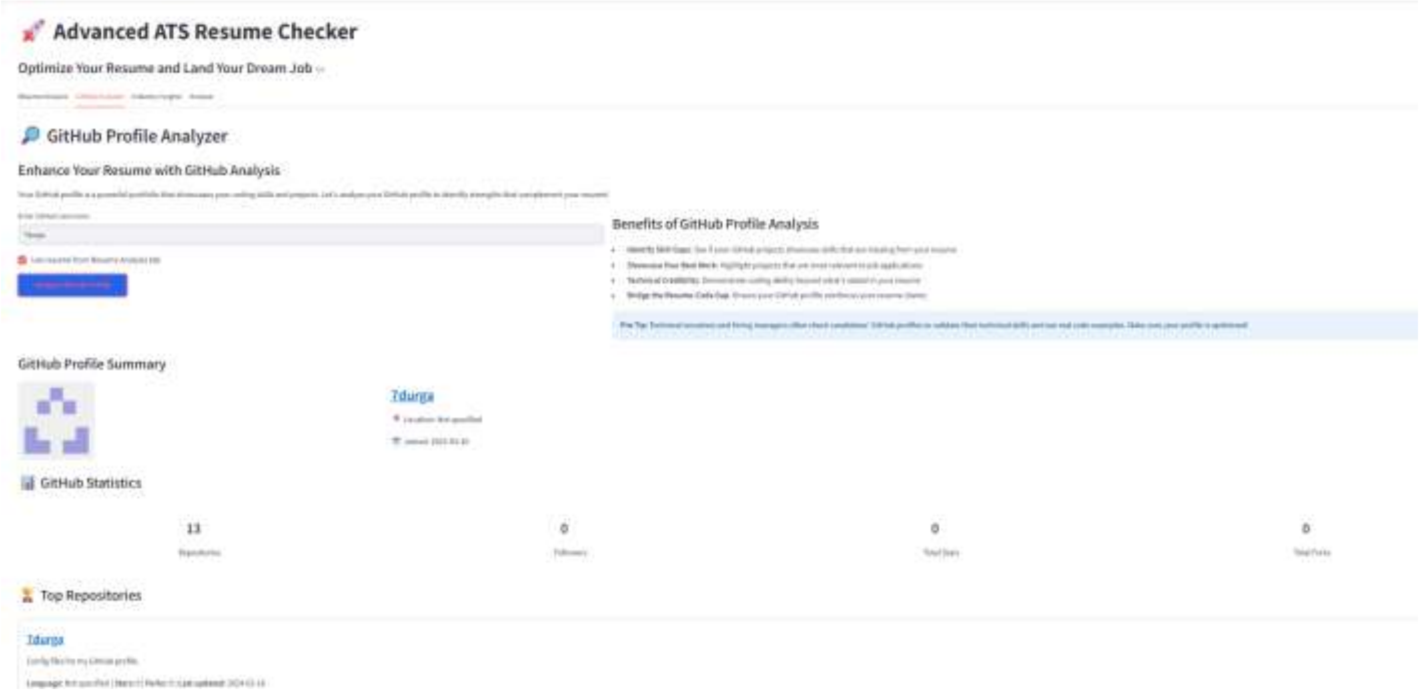
This resume scored 13.1% match with 13 keywords matched.

Top Performer	Highest Match	Keywords Matched
Durgaraokumili_resume (1).pdf	13.1%	13

Match Statistics

Average Match	Median Match	Min Match	Max Match
13.1%	13.1%	13.1%	13.1%

Visual Analysis



Advanced ATS Resume Checker

Optimize Your Resume and Land Your Dream Job

GitHub Profile Analyzer

Enhance Your Resume with GitHub Analysis

Benefits of GitHub Profile Analysis

GitHub Profile Summary

7durga

GitHub Statistics

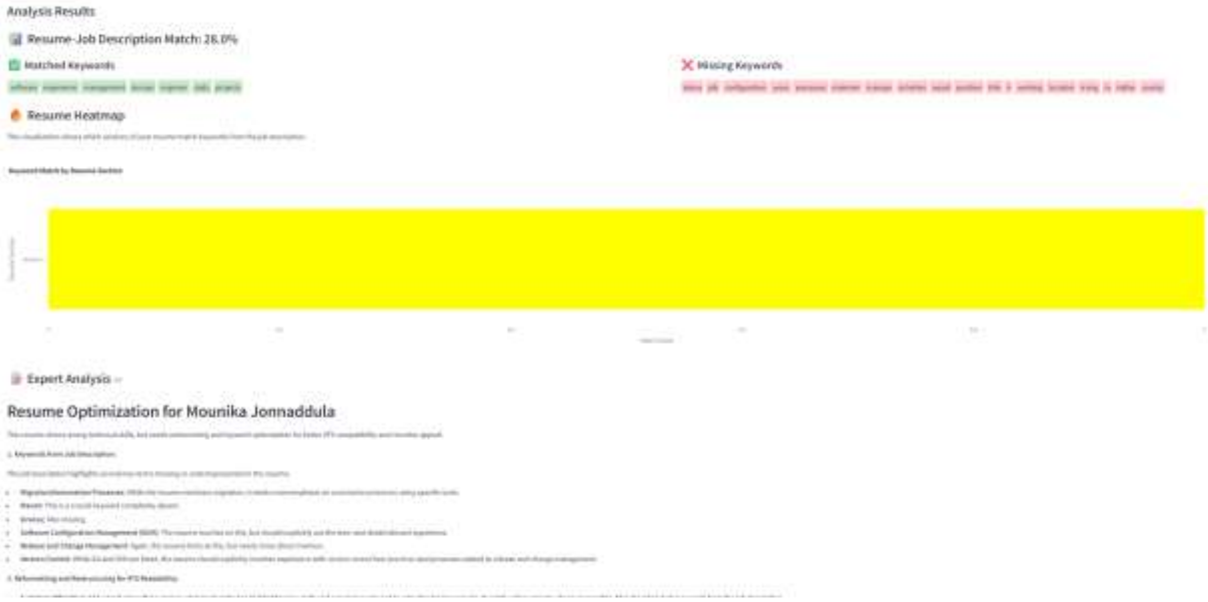
13	0	0	0

Top Repositories

7durga

4. RESULT:

The **Advanced ATS Resume Checker** delivers a comprehensive, data-driven output that enables job seekers to optimize their resumes for maximum ATS compatibility and relevance to targeted job descriptions. The system generates an **ATS score** that reflects the overall match between the resume and the provided job description, taking into account keyword presence, contextual relevance, and formatting compliance. Detailed **keyword match reports** highlight matched, partially matched, and missing skills, along with their importance weightings.



Analysis Results

Resume-Job Description Match: 28.0%

Matched Keywords

Missing Keywords

Resume Heatmap

Expert Analysis

Resume Optimization for Mounika Jonnaddula

5. DISSUSSION:

The **Advanced ATS Resume Checker** demonstrates how AI-driven analysis, combined with NLP and visualization techniques, can significantly improve the effectiveness of resume optimization for modern recruitment systems. Traditional resume writing often relies on guesswork, with job seekers unsure about whether their resumes align with the automated filters used by Applicant Tracking Systems (ATS). This platform addresses that gap by providing **objective, data-backed insights** into keyword relevance, formatting compliance, and skill alignment. One of the major strengths of the system is its **modular architecture**, which allows independent processing of resume parsing, keyword matching, formatting validation, and GitHub profile analysis

6. CONCLUSION :

This project successfully demonstrates the integration of machine learning algorithms with resume optimization and evaluation tools to improve the job-seeking process. The Advanced ATS Resume Checker provides users with an intuitive interface to compare resumes against job descriptions, analyze GitHub profiles for technical validation, and offer feedback to enhance employability. Additionally, the inclusion of OMR-based evaluation systems allows educators to grade and provide insights efficiently, leveraging automation and accurate data processing. The system enhances transparency, reduces manual errors, and empowers users with actionable insights. Overall, the Advanced ATS Resume Checker system bridges the gap between job seekers and the evolving expectations of modern recruitment technologies. By integrating intelligent resume analysis, GitHub profile evaluation, and feedback mechanisms, the platform empowers users to present their professional profiles more effectively. The system not only enhances the chances of passing through Applicant Tracking Systems but also helps individuals identify and improve gaps in their skills and experiences. This project stands as a scalable and practical solution in the domain of career readiness and talent assessment

7. FUTURE SCOPE:

In the future, this project can be enhanced by incorporating advanced Natural Language Processing (NLP) techniques to enable deeper semantic analysis of resumes and job descriptions, allowing for more precise matching. AI-powered resume improvement suggestions could be added to automatically enhance content and formatting based on industry trends. Integration with platforms like LinkedIn and job portals can provide richer profile data and broader application tracking. Additionally, developing a mobile-friendly interface would increase accessibility for users on the go. Real-time performance analytics dashboards for educators and administrators could offer actionable insights, while multilingual support would extend the system’s reach to a global audience. These enhancements would make the solution more intelligent, user-centric, and scalable.

8. ACKNOWLEDGEMENTS:

B.Kumari working as a assistant professor in master of computer application (MCA) sanketika vidya parishad engineeering college, viskhapatnam Andra pradesh.completed his post graduation in Andra university college of engineering (AUCE) with 2years of experience in computer science and engineering (CSE),accerdited by NAAC with his area of interst in java full stack.

Dharmireddy Manikanta is pursuing her final semester MCA in Sanketika Vidya Parishad Engineering College, accredited with A grade by NAAC, affiliated by Andhra University and approved by AICTE.with interest in neural networks chukka sowjanya has taken up her PG project on ““**CareerBoostAI: Resume And Github Optimization Platform”**

” and published the paper in connection to the project under the guidance of B. Kumari, Assistant Professor, SVPEC.

REFERENCES:

1) Recognition of pulmonary diseases from lung sounds using convolutional neural networks and long short-term memory
http://surl.li/yelyuc

2) A journal on machine learning in complex networks by Thiago Christiano Silvalinked http://surl.li/tufjak.

3) Lung sounds classification using convolutional neural networks https://www.sciencedirect.com/science/article/abs/pii/S0933365717302051.

4) A journal on effective auto grading with a webcam by Raveendran, Rahul Bharadwaj in ProQuest

http://surl.li/dnbjcm

5) A journal on the effectiveness of automation in evaluating test results using evalbee as an alternative optical mark recognition (OMR) by Michael jomar b. Ison in ijtaese

https://jurnal-fkip.ut.ac.id/index.php/ijtaese/article/view/661.

6) A journal optical mark recognition in student continuous assessment by journals & magazines by IEEE Xplore

https://ieeexplore.ieee.org/abstract/document/6926761.

7) A journal on hands-on image processing with Python by Sandipan dey

http://surl.li/twxczy

8) Classification of lung sounds using convolutional neural networks

https://link.springer.com/article/10.1186/s13640-017-0213-2

9) Lung disease classification using deep convolutional neural network https://ieeexplore.ieee.org/abstract/document/8983071

10) A journal on design patterns of deep learning with TensorFlow: building a customer hyper-personalization ecosystem using deep learning design patterns by Thomas v. Joseph linked: http://surl.li/yelyuc

11) A journal on machine learning in complex networks by Thiago Christiano Silvalinked:

http://surl.li/tufjak

12) A journal on effective auto grading with a webcam by Raveendran, Rahul Bharadwaj in ProQuest linked:

http://surl.li/dnbjcm

13) A journal on the effectiveness of automation in evaluating test results using evalbee as an alternative optical mark recognition (OMR) by Michael jomar b. Ison in ijtaese

https://jurnal-fkip.ut.ac.id/index.php/ijtaese/article/view/661

14) A journal optical mark recognition in student continuous assessment by journals & magazines by ieee xplore linked:

https://ieeexplore.ieee.org/abstract/document/6926761

15) A journal on hands-on image processing with Python by Sandipan dey linked:

http://surl.li/twxczy/

16) A journal on educational Ats Resume optimization in wikipedia

https://en.wikipedia.org/wiki/resume