Resume Parser AI Project

- **Under the Supervision of:**
  Dr. Reham
- **Teaching Assistant:**
  Eng. Ziad
- **Team Members:**
  Mohamed Alaa - Amr Hamouda - Mostafa Mohamed

Resume Parser AI Project

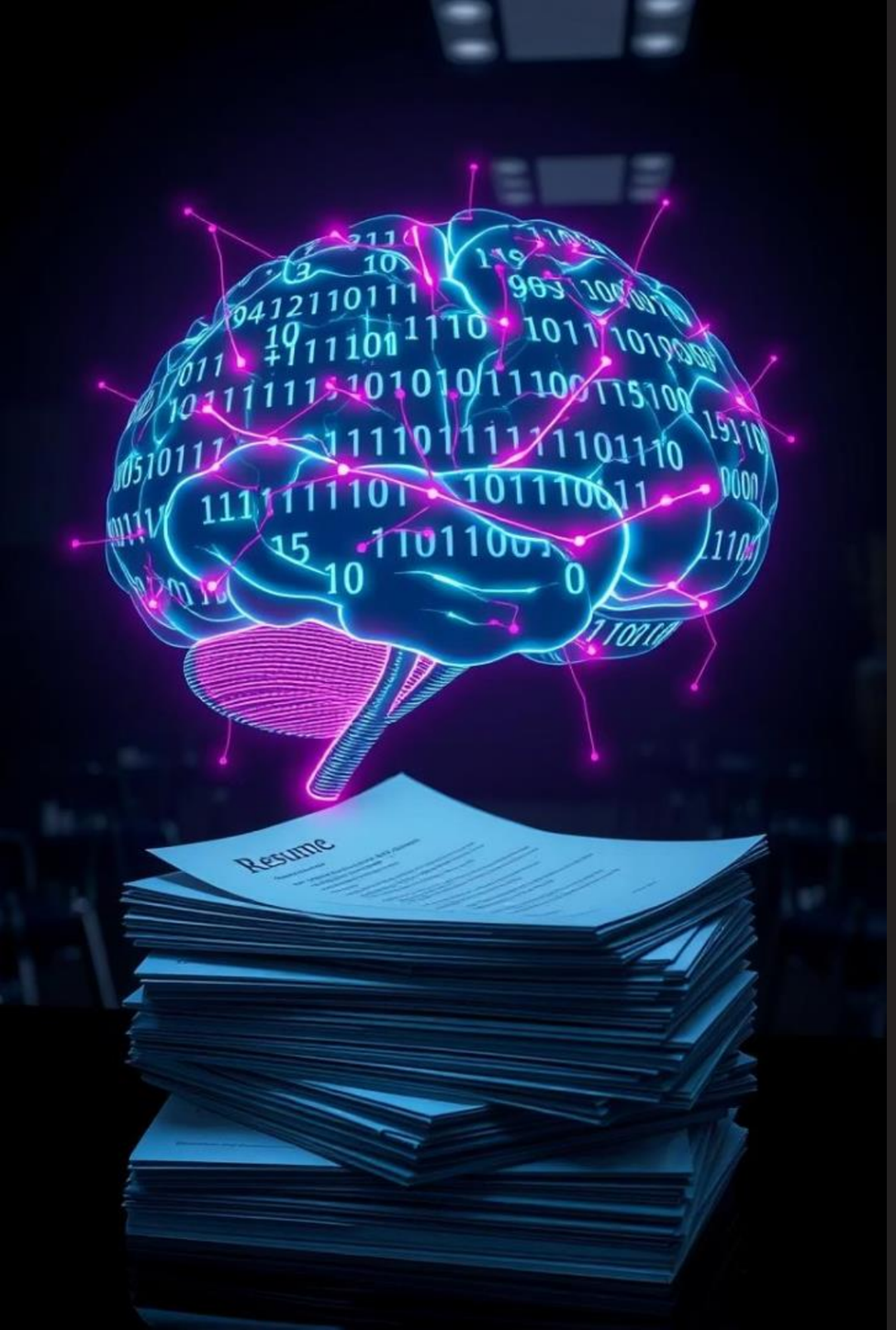
This presentation details the design and implementation of an AI-driven resume parser. The system intelligently screens resumes to identify the best candidates for job openings, aiming to automate and optimize recruitment processes. It combines advanced parsing techniques and AI scoring to streamline hiring decisions and improve recruiter productivity dramatically.

General Project Idea

This project is an intelligent resume screening system that automates candidate selection for job openings. By analyzing job requirements and parsing CVs, it identifies the best matches efficiently.

It eliminates the tedious manual review of hundreds of resumes by enabling automated skill extraction and candidate ranking based on AI-driven criteria.

- Automated resume parsing and data extraction
- AI-based skill and experience scoring
- Supports both English and Arabic resumes
- Designed for HR teams and recruiters



Project Objective

Efficient Automation

Reduce manual effort by auto-selecting the best candidates based on smart criteria.

Consistent Scoring

Apply data-driven, unbiased scoring methods to rank applicants uniformly.

Multilingual Support

Handle both Arabic and English resume content seamlessly, increasing inclusivity.

Why We Built It (Who Benefits?)

HR Departments

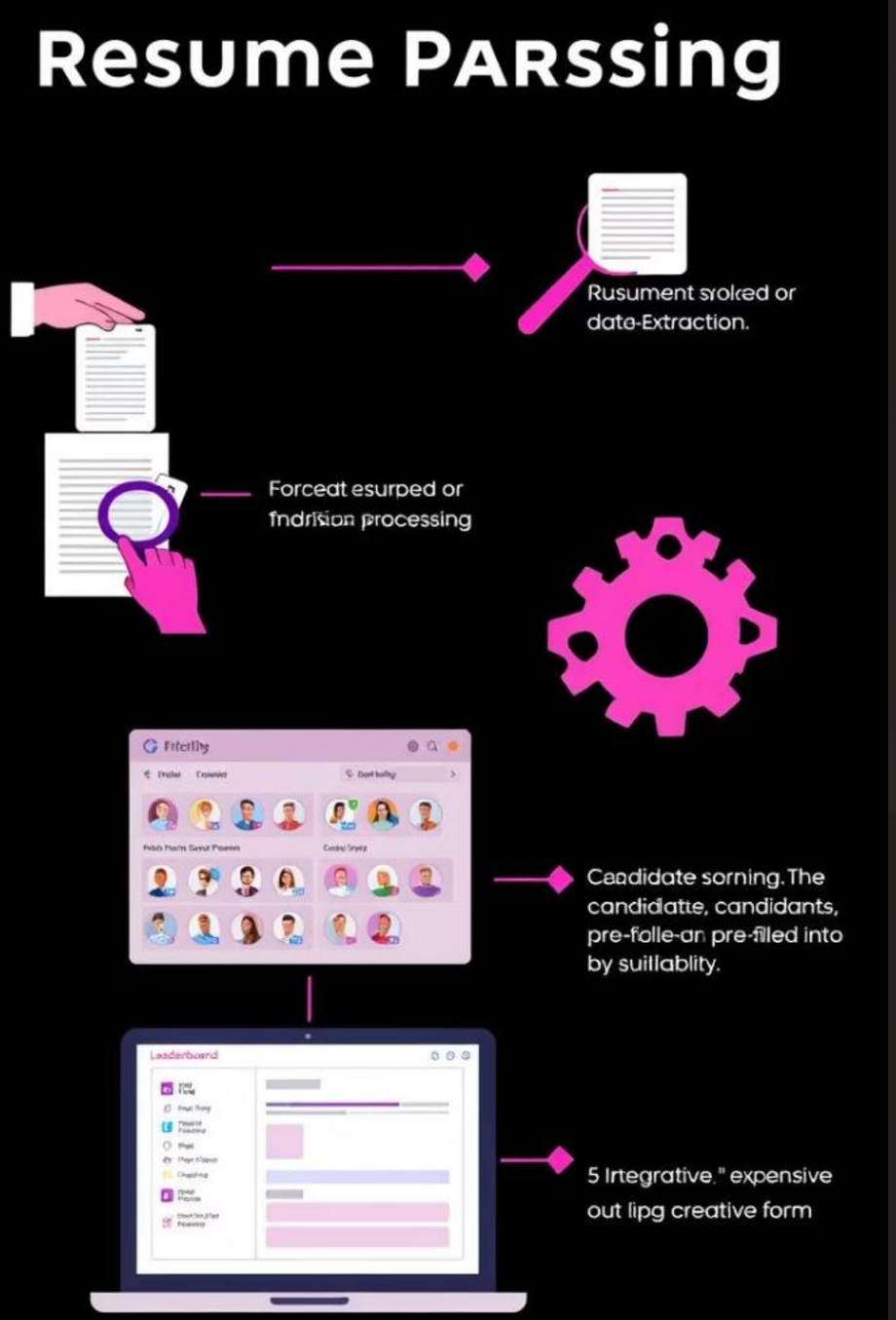
Automate screening to save valuable time and improve candidate selection accuracy.

Recruiters & Career Platforms

Offer advanced tools for smarter candidate matching as a competitive advantage.

Startups

Quickly identify top talent without dedicated hiring teams, enabling agile growth.



Project Workflow (Step by Step)

Input Job Requirements

User provides required skills in a simple comma-separated format to initiate parsing.

Resume Parsing

PDF and Word resumes are scanned to extract candidate information efficiently.

Skill Matching & Scoring

Extracted skills are translated if needed and fuzzy matched to the job requirements to assign scores.

Ranking & Reporting

Candidates are ranked based on scores, and results are saved with visual skill and experience charts.

Technologies & Libraries Used

Library	Purpose
pandas	Handles DataFrames and CSV file reading for candidate data management.
fuzzywuzzy	Performs approximate string matching for smart skill comparison.
python-docx	Extracts text from .docx resumes to process candidate information.
fitz (PyMuPDF)	Reads and extracts text from PDF format resumes.
matplotlib	Generates visual charts to display skills and experience for reports.

Code Structure Breakdown

1. `calculate_score(job_requirements, candidate_skills)`

- Matches candidate skills to job using fuzzy ratio
- +1 point per match over 80% similarity

2. `find_best_candidate(job_requirements, df)`

- Scores each candidate: skills, experience, communication
- Returns top scorer

3. `extract_info_from_text(text)`

- Extracts name, skills, experience, communication from resume text

4. `export_results()`

- Saves summary TXT and detailed CSV with scores

5. `visualize_stats()`

- Counts top 5 skills
- Creates experience histogram and skills bar chart

6. `main()`

- User inputs requirements
- Loads resumes and runs functions: extract → score → export → visualize

Output Example (Styled)

========== 🎯 Best Candidate Selected ==========

👤 Name : Ahmed Hassan

💡 Skills : python, sql, data analysis

📈 Experience : 5 years

🗣️ Communication: excellent

🏆 Score : 13

==========

This result is automatically saved in text and CSV files for record keeping. It offers a clear, visual summary to assist decision making and can be easily shared or further analyzed.

Contact Us

S220018@eng.ninu.edu.eg

Project Goals

We are excited about the opportunity to advance recruitment technology using AI and look forward to future developments.

Thank You