NeuraHire

NEURAHIRE

-Automate. Match. Hire.

Team Name: Alverse

Team Members:

1. Asawari Hire(Team Leader)
2. Yash Patil

NeuraHire

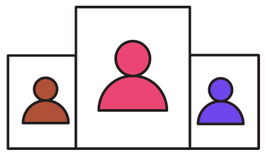
CONTENTS

Problem statement:

NeuraHire

In modern recruitment, HR professionals struggle with processing a high volume of resumes across various job domains. Manual evaluation is time-consuming and prone to human error and bias. The unstructured nature of resumes makes it difficult to extract key qualifications for accurate matching. As a result, suitable candidates may be overlooked, causing delays in hiring. An intelligent, automated system is needed to streamline everything from resume classification to shortlisting and interview scheduling.

Large number of unstructured resumes



Shortlisting resumes based on job domains

Filtering candidates who match 80% of job criteria

Inviting shortlisted candidates for interview via email.

Key Challenges:

- Slow manual resume screening
- Unstructured resume formats
- Hard to map resumes to JDs
- High risk of human bias
- Time wasted on poor matches
- Non-standard resume layouts
- Vague or inconsistent JDs
- No scalable hiring system
- No resume tracking mechanism
- Delayed shortlisting process
- Generic candidate communication
- No memory of past resume data

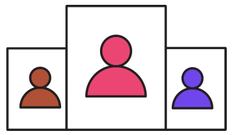
Proposed Solution Overview

NeuraHire

To tackle inefficiencies in traditional recruitment, we propose NeuraHire — a multi-agent AI system that automates and optimizes the hiring process. It begins by classifying resumes into relevant job domains using NLP. Structured data is extracted from resumes and JDs to compute a match score using AI-based similarity methods. Candidates scoring 80% or above are auto-shortlisted. The final agent sends personalized interview emails with proposed schedules. A centralized SQLite database supports long-term memory and traceability. This automation enhances hiring speed, accuracy, and fairness with minimal human intervention.

Resume Parser Agent

Tech: spaCy (NLP-based parsing)



Domain Classifier Agent

Tech: BERT / Sentence Transformers

JD Matching Agent

Tech: Cosine Similarity + SBERT

Email Scheduler Agent

Tech:SMTP + Jinja2 Templates

How NeuaHire solves problems?

- AI resume parser
- NLP-based structuring
- SBERT similarity scoring
- Bias-free AI matching
- Smart filtering (≥80%)
- Format-independent parsing
- JD summarization
- Scalable multi-agent system
- Resume data tracking (DB)
- Instant shortlisting
- Personalized email engine
- Long-term memory (SQLite)

Methodology

NeuraHire

Input

Upload Resume Files

Upload Job Descriptions

Parsing

Resume Parser - spaCy + PDF Reader

JD Summarizer - T5 Transformer

Extract Skills, Experience, Education

Extract JD Role, Skills, Requirements

Classify Domain - Zero-shot BERT

Normalize JD Keywords

Match Profiles - SBERT

Score Candidates - Cosine Similarity

Match ≥ 80%?

NO

Send Rejection email

YES

Send Email - SMTP

Calendar Invite - ICS

Sample Resume Files: CLICK HERE

Sample Job Description: CLICK HERE

Technologies and Frameworks

NeuraHire

1. Resume Parsing Agent
Purpose- Extract structured data from resume

1

Tools & Frameworks Used:
- spaCy – NLP to identify named entities (skills, orgs, degrees)
- PyMuPDF / pdfminer.six – PDF reading & text extraction
- SkillNer (optional) – For extracting domain-specific skills
- Custom Regex / Synonym Mapper – Normalize skill names

Tools & Frameworks Used:
- HuggingFace Transformers
- → Models: T5, BART, or DistilBART for summarization
- Zero-shot Classifier (BART-MNLI)
- → Categorize JD into job domain without training data
- spaCy / NLTK – Light text pre-processing
- Skill Normalizer – For mapping JD skills to standard tags

2

2. Domain Classifier Agent
Purpose: Summarize job descriptions and extract key requirements.

3. JD Matching Agent
Purpose: Match resumes to JDs and calculate match scores.

3

Tools & Frameworks Used:
- Sentence-BERT (SBERT) – Semantic similarity between resume & JD
- Scikit-learn – Scoring logic, thresholding, normalization
- LIME (optional) – Explainable AI for understanding scoring logic
- Pandas / NumPy – Data handling and computations

4

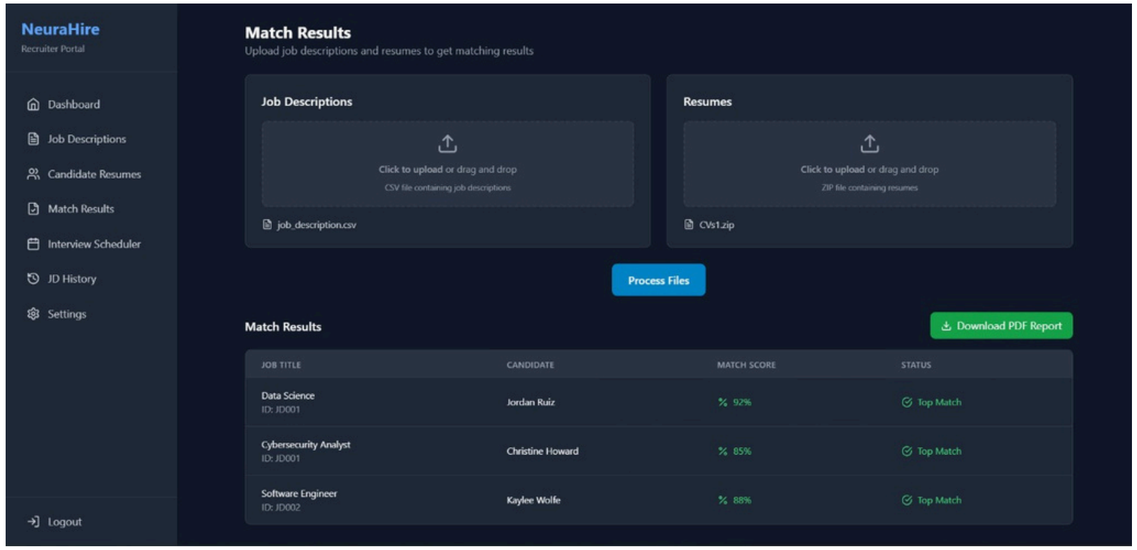
4. Email Scheduler Agent
Purpose: Send selection/rejection emails and interview invites.

Tools & Frameworks Used:
- smtplib / yagmail – Sending emails via SMTP
- ics.py – Generate .ics calendar invite files
- Datetime / Timezone – Format interview timings properly
- SQLite – Store scheduling history and email logs

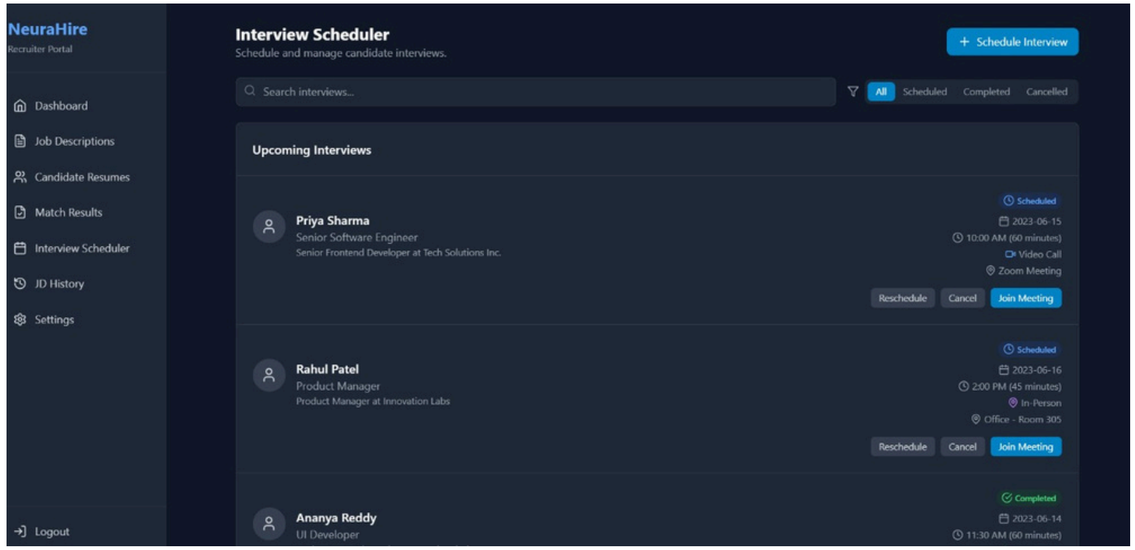
Recruiter Interaction Interface

NeuraHire

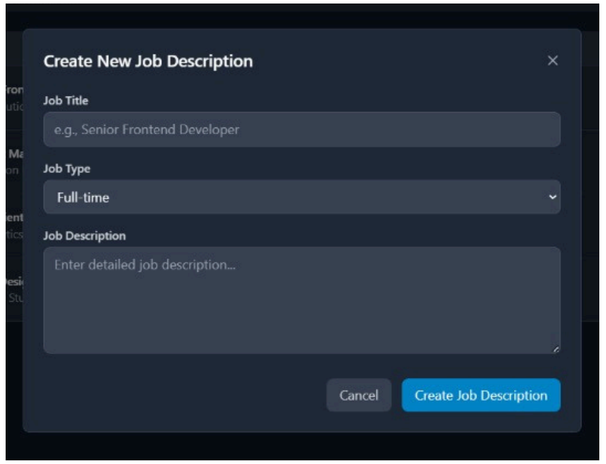
1. Input: Upload Job Descriptions and Resumes received



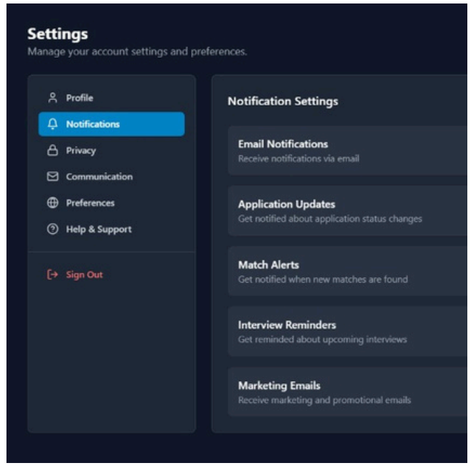
2. Interview Scheduling.



3. Creating a new Job Description



4. Notifications and Alerts



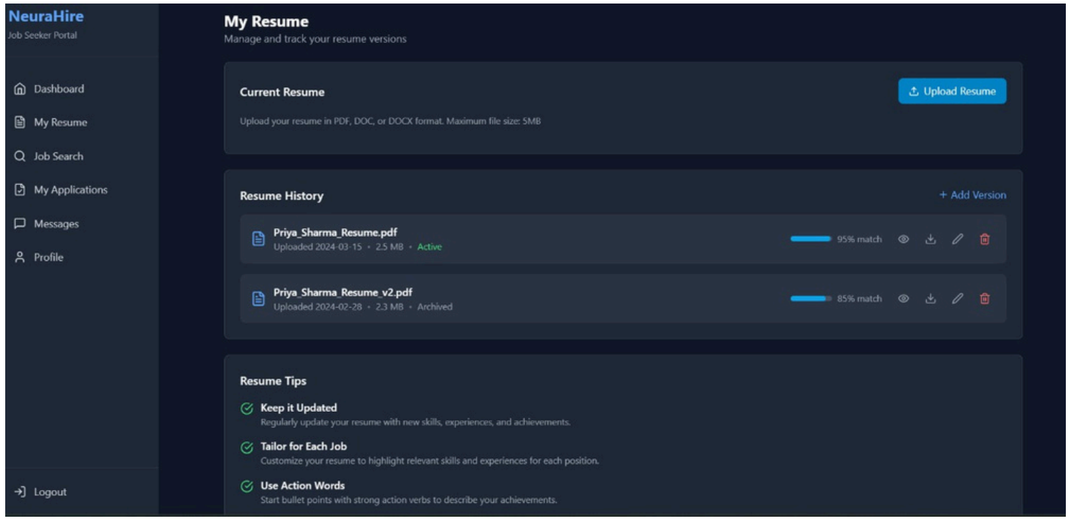
For Recruiter Interface Demo

Click here

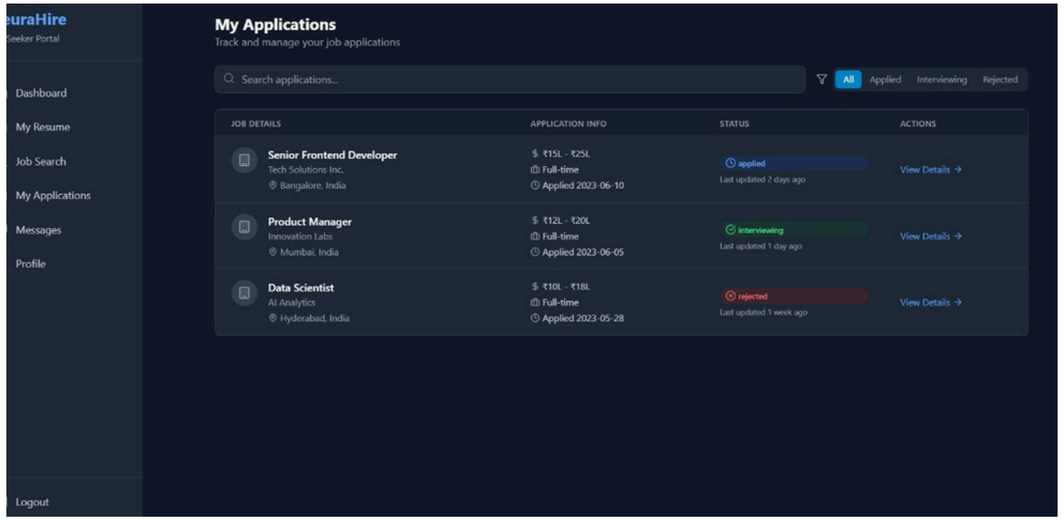
Candidate Interaction Interface

NeuraHire

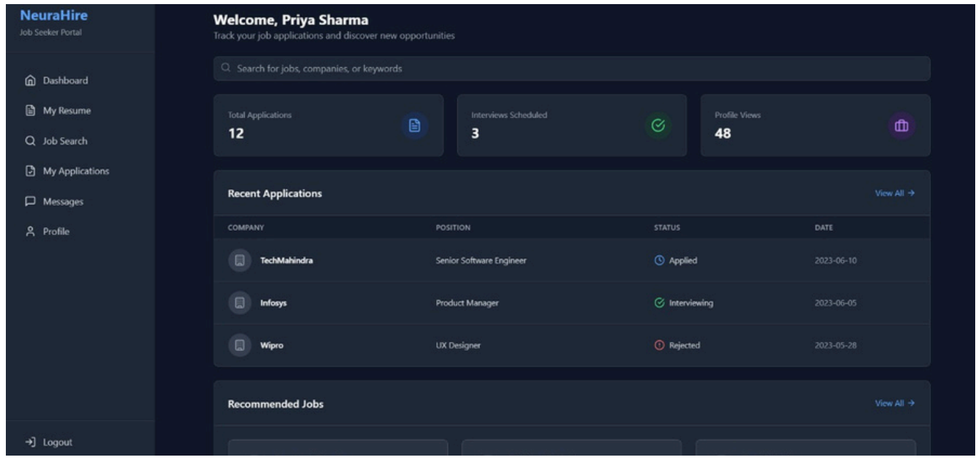
1. Candidate Resume(Input), Resume History and Tips



2. Applied Positions, Status and Application info



3. Companies Applied ,Interviews Scheduled and Profile Views



For Candidate Interface Demo

Click here

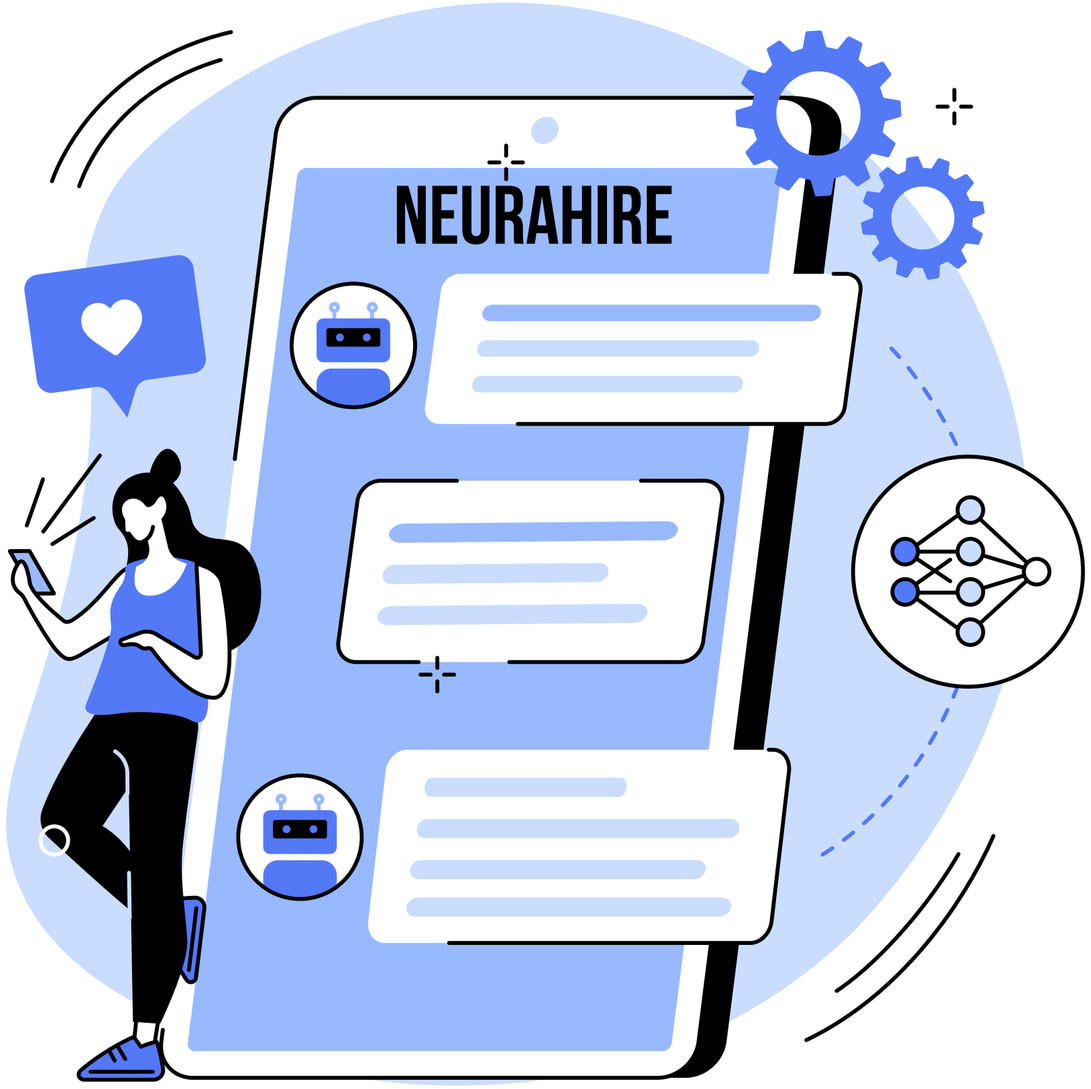
Comparative Analysis

NeuraHire

Aspect	Traditional Hiring	NeuraHire
Resume Screening	Manual, slow, error-prone	Automated, fast, consistent
Resume Format Handling	Struggles with unstructured formats	Parses and structures resumes using NLP
Job Description Matching	Keyword-based, often inaccurate	Semantic understanding & intelligent matching
Time Efficiency	Takes days or weeks to process	Processes in minutes
Bias & Fairness	Subject to human bias	Data-driven, unbiased scoring
Scalability	Hard to scale with large volume	Easily handles hundreds/thousands of resumes
Shortlisting Logic	Often based on intuition or rigid filters	Uses 80% match score threshold with explainable logic
Communication	Generic or delayed emails	Sends personalized, instant interview invitations
Candidate Experience	Delayed response, unclear feedback	Timely updates, structured process

NEURAHIRE

NeuraHire



THANK YOU

-Here’s to an Effortless Hiring Journey