

AI Resume Screening and Ranking System

Overview

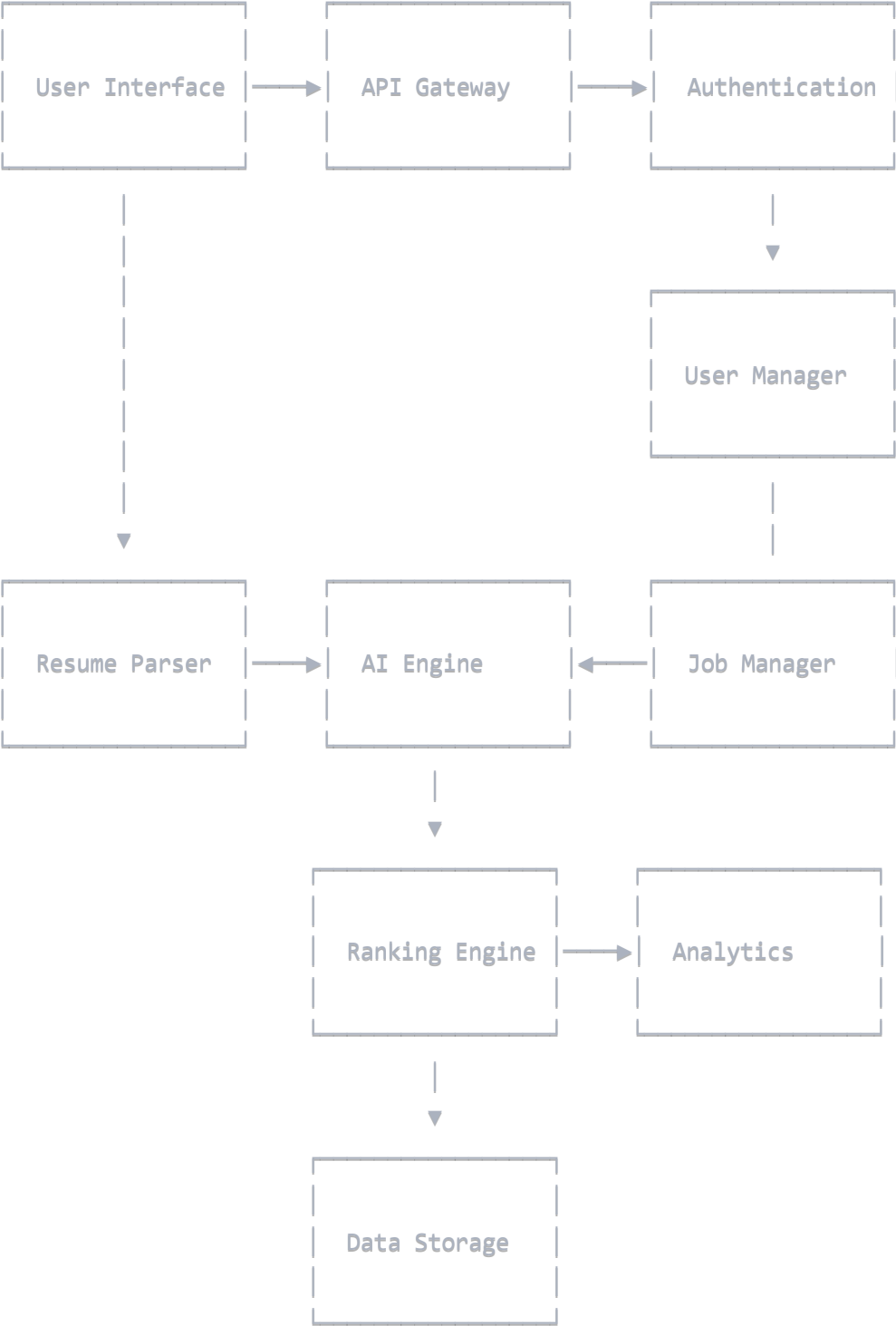
The AI Resume Screening and Ranking System is an intelligent solution designed to streamline the recruitment process by automatically analyzing, screening, and ranking resumes based on job requirements. This system reduces the manual effort of HR professionals and hiring managers while improving candidate selection efficiency and accuracy.

Features

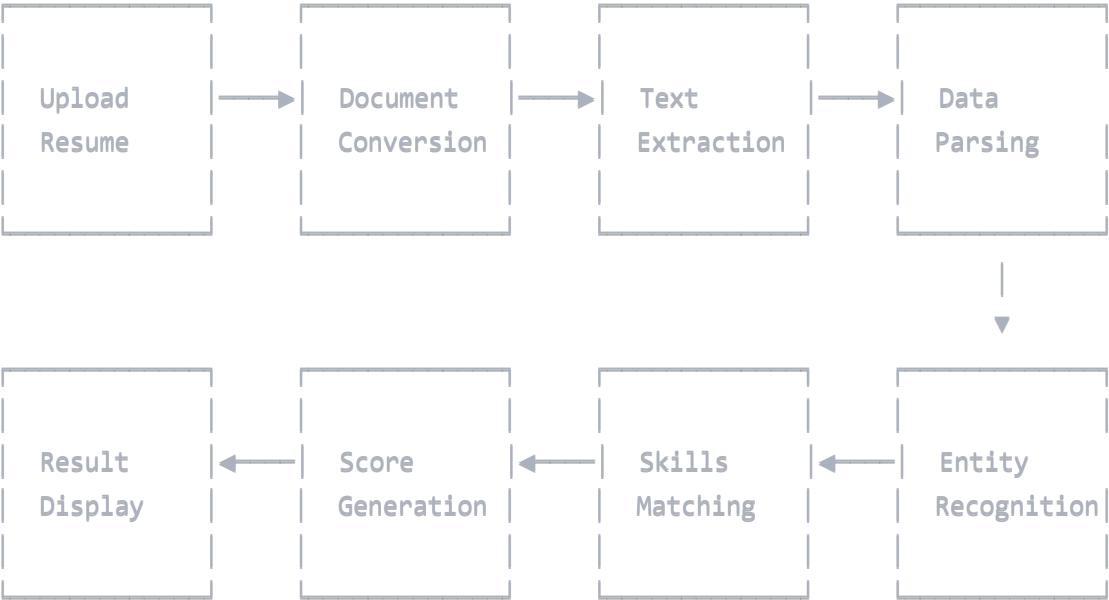
- **Automated Resume Parsing:** Extracts key information from resumes in various formats (PDF, DOCX, TXT)
- **Intelligent Matching:** Compares candidate qualifications against job descriptions using NLP
- **Custom Ranking Algorithm:** Ranks candidates based on multiple weighted criteria
- **Bias Mitigation:** Implements techniques to reduce unconscious bias in the selection process
- **Interactive Dashboard:** Provides clear visualization of candidate rankings and comparisons
- **Batch Processing:** Handles large volumes of resumes efficiently
- **Export Functionality:** Generates reports in various formats (CSV, PDF, Excel)

System Flowcharts

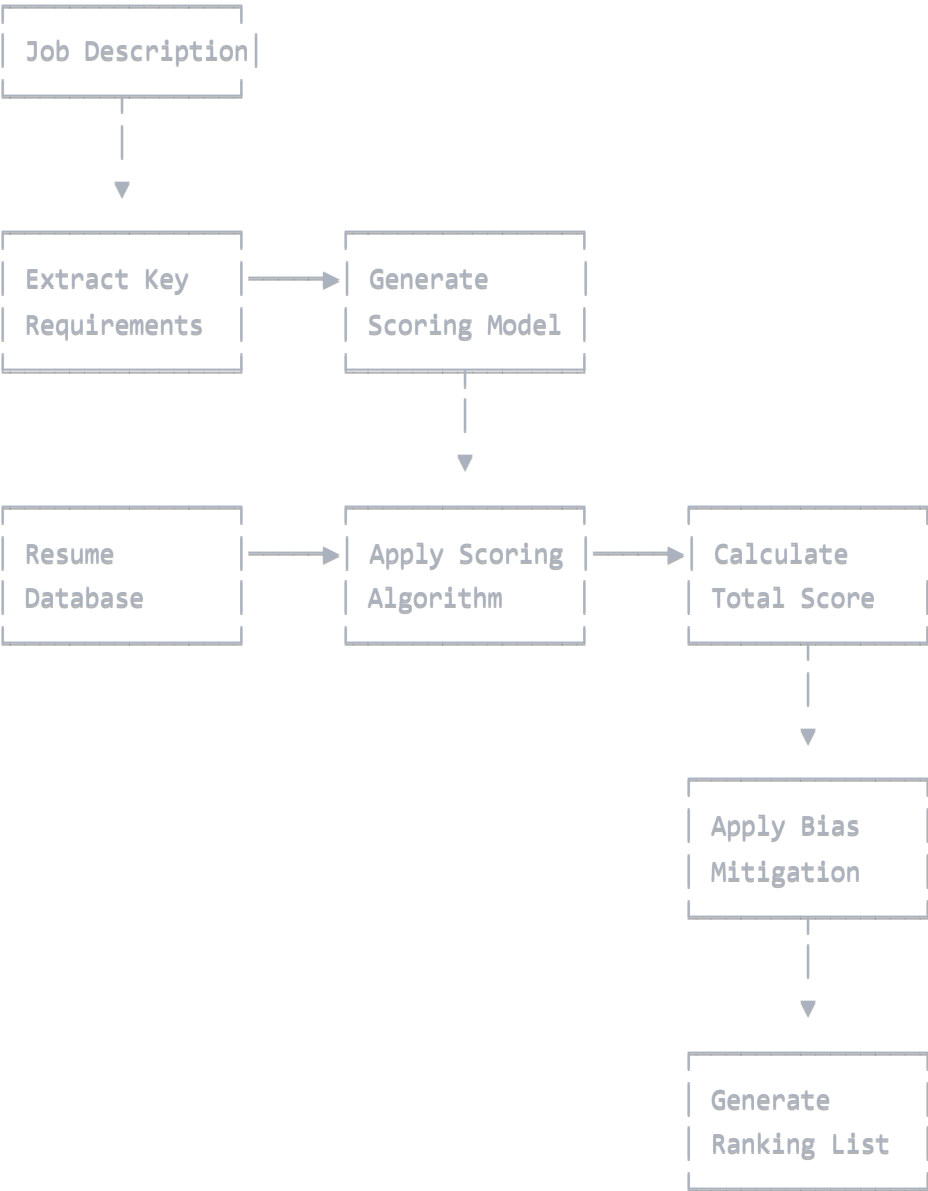
High-Level System Architecture



Resume Processing Workflow



Candidate Ranking Process



Technology Stack

- **Backend:** Python, Flask/FastAPI
- **Frontend:** React.js, Material-UI
- **AI/ML:** TensorFlow/PyTorch, NLTK, spaCy
- **Database:** MongoDB/PostgreSQL
- **Deployment:** Docker, AWS/GCP

Installation

Prerequisites

- Python 3.8+
- Node.js 14+
- MongoDB/PostgreSQL
- Docker (optional)

Setup Instructions

1. Clone the repository

```
bash
```

```
git clone https://github.com/yashchavan5/ai-resume-screening.git
cd ai-resume-screening
```

2. Set up the backend

```
bash
```

```
cd backend
python -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate
pip install -r requirements.txt
```

3. Set up the database

```
bash
```

```
# Follow database-specific setup instructions in /docs/database-setup.md
```

4. Set up the frontend

```
bash
```

```
cd ../frontend  
npm install
```

5. Configure environment variables

```
bash
```

```
cp .env.example .env  
# Edit .env with your configuration
```

6. Run the application

```
bash
```

```
# Start backend
```

```
cd backend  
python app.py
```

```
# Start frontend in a new terminal
```

```
cd frontend  
npm start
```

Usage

Basic Usage

1. Login to the dashboard
2. Upload job description
3. Upload resumes (batch upload supported)
4. Initiate the screening process
5. View ranked results
6. Export or share findings

Advanced Features

- **Custom Criteria Weighting:** Adjust importance of different qualification factors
- **Keyword Configuration:** Add domain-specific keywords for better matching
- **Anonymization Options:** Remove identifying information to reduce bias
- **API Integration:** Connect with ATS and HRIS systems

Documentation

Comprehensive documentation is available in the `/docs` directory:

- [User Guide](#)
- [API Documentation](#)
- [Admin Guide](#)
- [Developer Documentation](#)

Contributing

We welcome contributions to improve the AI Resume Screening System!

1. Fork the repository
2. Create your feature branch (`git checkout -b feature/amazing-feature`)
3. Commit your changes (`git commit -m 'Add some amazing feature'`)
4. Push to the branch (`git push origin feature/amazing-feature`)
5. Open a Pull Request

Please read [CONTRIBUTING.md](#) for details on our code of conduct and submission process.

Roadmap

- Integration with popular job boards
- Enhanced candidate communication features
- Video interview analysis
- Skills assessment integration
- Mobile application

License

This project is licensed under the MIT License - see the [LICENSE](#) file for details.

Contact

Yash Chavan - yashchavanpatil1@gmail.com

Project Link: <https://github.com/yashchavan5/ai-resume-screening>

Acknowledgments

- All open-source libraries used in this project
- Contributors and testers
- Feedback from HR professionals