SHL Assessment Recommendation Engine

Objective

Develop an SHL Assessment Recommendation Engine to suggest suitable assessments based on job roles and skill levels. This tool helps recommend relevant SHL assessments for users, ensuring an appropriate match between assessments and job criteria.

Tech Stack

- **Python**: Backend development
- **Streamlit**: Web app framework
- **Pandas**: Data manipulation
- **Flask**: API service
- **GitHub**: Version control

Data

The `assessments.csv` dataset includes:

- **Assessment_ID**: Unique identifier
- **Assessment_Name**: Assessment title
- **Job_Role**: Relevant job roles (e.g., Analyst, Manager)
- **Skill_Level**: Skill levels (e.g., Entry, Mid, Senior)
- **Tags**: Relevant tags (e.g., cognitive, leadership)

Steps Taken

1. **Data Preprocessing**: Cleaned the dataset to focus on job roles and skill levels.

2. **Recommendation Logic**: Implemented rule-based filtering of assessments by job role, skill level, and tags.

3. **Streamlit Interface**: Built a web interface where users input job role and skill level to get recommendations.

4. **Flask API**: Created an API that returns recommendations in JSON format based on job role and skill level.

5. **Testing & Deployment**: Tested the system for accuracy, and deployed locally.

6. **Version Control**: Used Git for version management and pushed updates to GitHub.

Future Improvements

- **NLP Integration**: Use BERT or spaCy for semantic understanding of tags.
- **API Integration**: Integrate with real SHL product APIs for dynamic data.
- **Enhanced UI**: Improve the Streamlit interface with more advanced filters.

GitHub Repository

shl_recommender

Conclusion

This recommendation engine is a simple yet effective solution for suggesting SHL assessments. It can be extended with advanced features such as NLP and real-time data integration