Career Recommendation System

Ch. Tharun[1], V. Srilekha[2], T. Maanasa[3], M. Sree Priya[4], M. Kavya[5]
[1]*Associate Professor, Hyderabad Institute of Technology and Management, Medchal, Telangana*
[2,3,4,5]*UG Student, Hyderabad Institute of Technology and Management, Medchal, Telangana*

***Abstract*—This paper proposes a novel career recommendation system that integrates Natural Language Processing (NLP) techniques powered by the SpaCy library to provide personalized career advice. The system collects detailed user inputs encompassing skills, interests, academic background, personality types, and career aspirations through a web-based interface developed with Python Flask. Utilizing semantic similarity measures between embeddings of user inputs and a structured career keyword dataset, the system overcomes limitations of rigid keyword matching by understanding the contextual meaning of natural language, including synonyms and abbreviations. The hybrid approach also incorporates rule-based filters aligning with academic streams and personality traits to refine suggestions. Experimental results demonstrate improved recommendation accuracy and user satisfaction, highlighting the system's potential as an adaptive career counseling tool.**

I. INTRODUCTION

The complexity of modern job markets and varied career pathways makes suitable guidance a critical need, especially for students and early-career professionals. Existing career recommendation systems often rely on static, keyword-based matching methods that inadequately capture the nuances of free-text user inputs or fail to personalize suggestions deeply. Natural Language Processing (NLP) has emerged as a transformative technology enabling machines to interpret and analyse human language in a meaningful way.

This project leverages NLP to analyse semantically rich user data, extracting intent and context to offer precise career pathways. By integrating a hybrid semantic similarity and rule-based filtering approach, the system addresses individual differences, adapts to diverse user inputs, and provides dynamic career guidance through an accessible web interface. This paper outlines the system's design, implementation, testing, and potential for future expansion.

II. LITERATURE SURVEY

Reviewing the evolution of career recommendation systems reveals a trajectory from simple rule-based models to complex AI-integrated frameworks. Initial systems depended heavily on fixed attribute matching, which limited flexibility and failed with ambiguous or descriptive user queries. Recent approaches apply collaborative and content-based filtering techniques from machine learning to improve personalization but require extensive labelled data and face explainability challenges. Advances in NLP, particularly with word embeddings and transformer-based models, facilitate semantic understanding that allows systems to handle synonyms, paraphrases, and context-dependent language. Hybrid models incorporating semantic similarity with domain-specific rules have shown superior performance in tailoring recommendations and enhancing user engagement. This project builds upon these advances by deploying SpaCy's efficient embeddings combined with explicit filtering to tailor career advice in real time.

III. METHODOLOGY

The methodology comprises a structured workflow to build a scalable and effective career recommendation engine:

- Requirement Analysis: Detailed study to gather functional and non-functional specifications establishing system capabilities and constraints.
- System Design: Adoption of a layered architecture isolating presentation, application logic, and data processing to promote modularity and maintainability.
- Data Acquisition: Collection of user data through multilingual-friendly input forms capturing academic records, technical and soft skills, personality profiles, and career goals.
- Preprocessing: Use of SpaCy for tokenization, lemmatization, named entity recognition, and

vector embedding extraction to convert input text into numerical semantic representations.

- Similarity Computation: Implementation of cosine similarity algorithms to measure distances between user input embeddings and curated career category vectors from a professional job dataset.
- Rule-Based Filtering: Development of heuristics incorporating academic streams, personality types, and industry sector relevance to refine career choices.
- Recommendation Generation: Aggregation of similarity scores and rule evaluations to generate ranked and categorized career paths tailored to individual profiles.
- Deployment: Integration within a Flask web server offering real-time, user-friendly interaction and dynamic result rendering.

IV. IMPLEMENTATION

The career recommendation platform is implemented using:

- Frontend: HTML5 forms styled with CSS3 provide accessible data entry points. Flask's Jinja2 templating engine enables dynamic rendering of recommendations.
- Backend: The Flask framework supplies robust routing, handles form submissions, and orchestrates interaction between user inputs and the processing engine.
- NLP Processing: SpaCy's medium English model (en_core_web_md) is integrated to convert raw text inputs into context-aware embeddings.
- Similarity Engine: Python scripts compute cosine similarities between user embeddings and predefined career keyword vectors stored in CSV/JSON files.
- Rule Engine: Custom heuristics apply business rules based on academic and personality data. The combined results are formatted and sent back for user display.
- Testing: Multiple unit and integration tests validate each module's performance. User acceptance testing with diverse input cases ensures real-world reliability.

V. SOFTWARE COMPONENTS

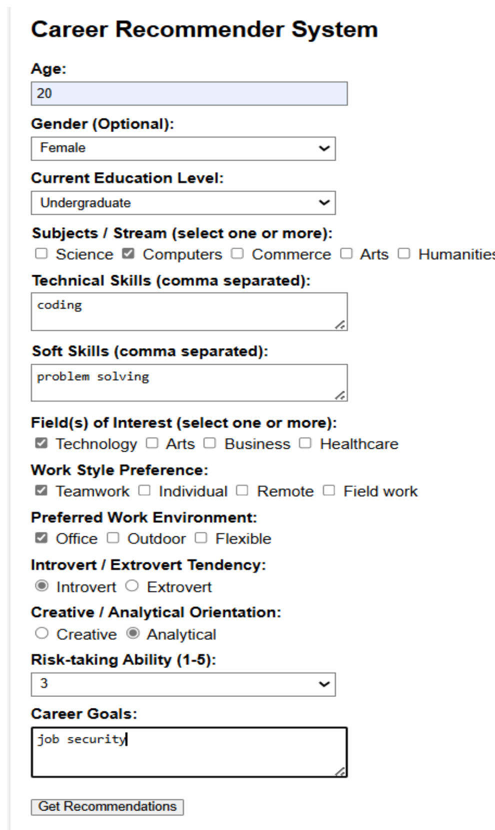
- Flask Framework: Handling HTTP requests, session management, and rendering dynamic web responses.
- SpaCy NLP Library: Natural language processing tools for semantic understanding.
- Frontend Technologies: HTML, CSS, and Flask templates for UI presentation.
- Data Files: Profession and skill keyword sets stored in JSON/CSV formats for lookup and comparison.
- Python Scripts: Core logic for similarity computation, rule filtering, and recommendation generation.

VI. RESULTS

Extensive testing demonstrates high accuracy in interpreting a wide range of user inputs including abbreviations, synonyms, and complex phrases and matching them with appropriate careers. The hybrid semantic-rule model shows improved recommendation relevance compared to ontology or pure keyword systems. Feedback from test users highlights increased satisfaction due to personalized, context-aware suggestions delivered promptly via an intuitive interface. Performance metrics indicate efficient computation supporting scalable, real-time deployment.

The output comes like this:

This is the User Interface



These are the recommendations after pressing “Get Recommendations” in the User Interface:

VII. CONCLUSION

The presented NLP-based career recommendation system offers a significant advancement over traditional static, keyword-centric tools by embedding semantic understanding and personalized logic within a web application framework. Its modular design promotes extensibility, allowing future incorporation of multilingual support, machine learning classifiers, and labour market integration. The system effectively supports users in navigating complex career decisions through intelligent, flexible, and accurate recommendations.

REFERENCES

[1] Shuyun S. Vignesh et al. (2021), Intelligent Career Guidance System using Machine Learning.
[2] Tanya V. Yadalam et al. (2020), Career Recommendation Systems using Content-Based Filtering.
[3] J. Joseph et al. (2019), Career Recommendation System Design using Machine Learning Technique.
[4] Lokam et al. (2025), AI-Based Career Guidance System.
[5] SpaCy Documentation, https://spacy.io
[6] Flask Documentation, https://flask.palletsprojects.com
[7] Career Recommendation System, IJIRT.
[8] Job Recommendation System Using NLP, IJRASET.