Project Proposal: Job Recommendation Engine for SEEK

Biying Chen

1. Background

In the dynamic job market, SEEK targets to bring the right positions to the right people. To realise this ambition, SEEK is building a job recommendation engine to increase the view and application rates of the jobs on the market by:

- Accurately predicting a job seeker's likelihood to **view** and **apply** for a job.
- Ranking recommendations based on **relevance** and predicted application **likelihood**.

Our goal is to achieve a 10–15% increase in application and view rates through personalized recommendations.

2. Data Sources & Tools

Data sources

- Job Ads
    - Size: 50k records
    - Format: JSON file (semi-structured, with missing values, outliers)
    - Field scope: Description, Title, Location, Salary, Work type, Classification, etc.
- Job Event
    - Size: 4.3M records on 2021-05-01
    - Format: csv file (structured, with duplicates)
    - Field scope: Timestamp, Resume_id, Platform, Kind, etc.

Tools

- Python processing and packages
    - Data exploration and cleaning: *pandas*, *nltk* (for text)
    - ML modeling: *sklearn*, *tensorflow* (for Neural Network)
    - Data visualization: *matplotlib, seaborn*
- Google Colab
    - Free online IDE
    - Model training
- GitHub
    - version control
- Google Cloud Platform
    - pipeline implementation

3. Proposed Solution

A. Hybrid Recommendation Approach

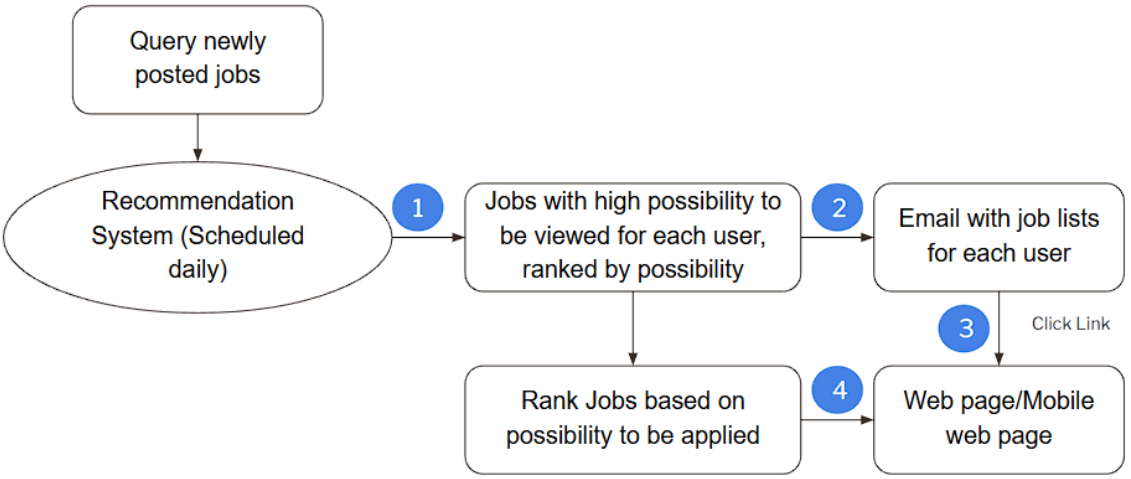
- **Predictive Models:** Binary classification (VIEW likelihood and APPLY likelihood) using **Random Forest** and **Neural Networks** .

B. Feature Engineering

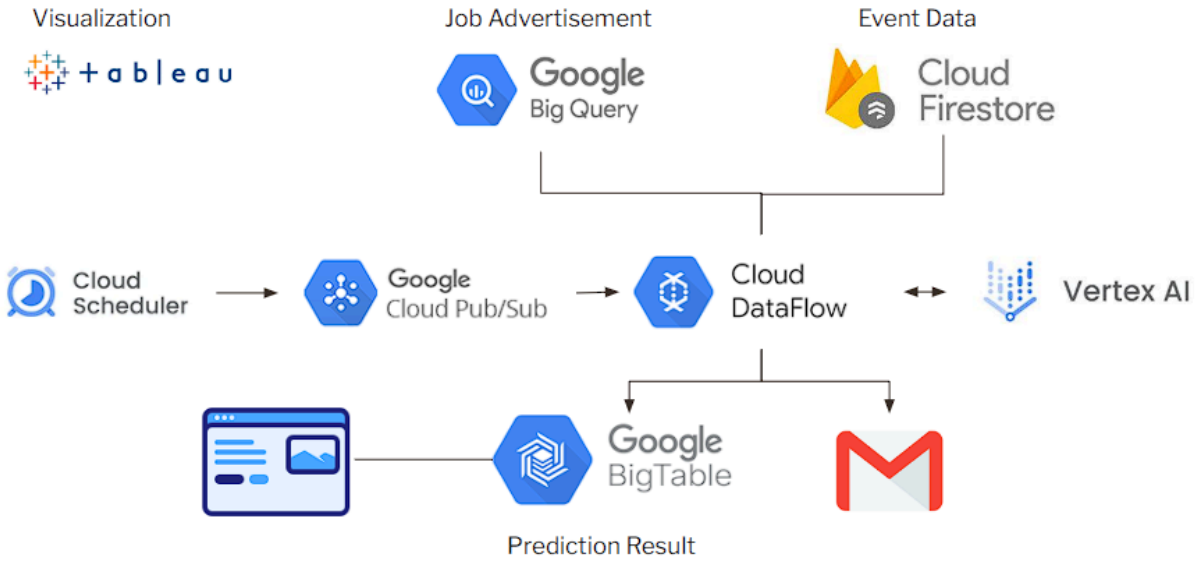
- Remove duplicated rows
- Add coordinate info through external API
- Text features (via TfidfVectorizer and regex).
- Aggregated metrics (job location and distance to applicants).

C. Pipeline

- **Functional Pipeline**



- **Technical Pipeline**



4. Evaluation & Validation

Metrics

- Model performance
    - **F1-score** (balance precision and recall rate)
- Business impact
    - Short-term: A/B test **application rates** and **view rates**
    - Long-term: (for employee) Using cohort to track **user retention rate** before and after releasing the recommendation engine, (for employer) Conducting NPS survey to collect **satisfaction rates**

Risks Mitigation

- Fallback: If the model F1-score< 0.5 (a predefined threshold), use heuristic rule instead.
- Bias checks: Keep monitoring recommendations for fairness across demographics.

5. Implementation Plan

This implementation plan should be feasible technically but may not perfectly be aligned with SEEK's tech stacks.

Phase	**Deliverables**
Architecting data pipeline	Draw pipeline architecture Diagram, develop ETL workflow.
Develop data models	Model logic documentation, Model training and evaluation.
Deployment	Deploy CI/CD pipelines on Jenkins, code on Bitbucket.
Monitoring	Integrated with data visualization tool Tableau/Power BI, monitor real-time dashboard.

6. Expected Outcomes

Outcomes	**Success metric**	**Validation Method**
Application/View Rate increase	15%, vs. status quo	A/B testing
Candidate retention increase	10% increase in D7, D14, D30 retention rates	Cohort analysis
Employer satisfaction rate increase	20% increase in NPS score	NPS survey