

1. 연봉 예측 모델

목표

- 주어진 채용 공고의 속성(경력, 직무, 지역, 학력 등)을 기반으로 연봉을 예측합니다.

데이터

- 입력 변수: 경력 수준, 직무, 지역, 학력 수준, 회사 규모, 산업 분야 등
- 출력 변수: 연봉 (숫자형, 회귀 문제)

알고리즘

- 회귀 모델: 선형 회귀, 다중 회귀, 랜덤 포레스트 회귀, XGBoost, LightGBM, 딥러닝 (ANN)
- 하이퍼파라미터 튜닝: 그리드 서치, 랜덤 서치
- 평가 지표: RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), R^2

구현 방법

- 입력 변수에서 중요한 특성(피처)을 선택하고, 머신러닝 알고리즘을 이용해 연봉 예측 모델을 학습시킵니다.
- 특성 엔지니어링을 통해 데이터 전처리와 변환을 고려하고, 교차 검증을 통해 모델의 성능을 평가합니다.

추천 도구

- Python:** Pandas, Scikit-learn, XGBoost, LightGBM, TensorFlow/Keras (딥러닝)
- 데이터 시각화: Matplotlib, Seaborn

2. 직무 분류 모델

목표

- 주어진 채용 공고를 자동으로 직무별로 분류합니다.

데이터

- 입력 변수: 채용 공고의 제목, 회사명, 지역, 요구 경력, 요구 기술 스택, 직무 설명 등
- 출력 변수: 직무 카테고리 (예: 소프트웨어 개발, 데이터 분석, 마케팅 등)

알고리즘

- 분류 모델: 로지스틱 회귀, **SVM** (서포트 벡터 머신), 랜덤 포레스트, **XGBoost**, 딥러닝 (**CNN**, **RNN**)
- 텍스트 전처리: 자연어 처리(**NLP**) 기법 사용 (예: TF-IDF, Word2Vec, BERT)
- 모델 평가 지표: 정확도, **F1-score**, Precision, Recall, ROC AUC

구현 방법

- 텍스트 처리: 채용 공고의 텍스트를 전처리한 후, 벡터화(**TF-IDF** 또는 **Word2Vec**)하여 모델에 입력합니다.
- 모델 학습: 분류 모델을 사용해 직무 카테고리를 예측하도록 합니다.
- 학습 및 평가: 데이터를 훈련/검증/테스트 세트로 분할하여 학습하고, 성능을 평가합니다.

추천 도구

- **Python**: Scikit-learn, TensorFlow/Keras (딥러닝), NLTK, SpaCy
- 데이터 시각화: Matplotlib, Seaborn

3. 구직자-직무 매칭 시스템

목표

- 구직자의 정보(경력, 기술, 학력 등)와 채용 공고의 정보(직무, 기술 스택 등)를 비교하여 매칭합니다.

데이터

- 입력 변수: 구직자 프로필(경력, 기술, 직무 선호도 등), 채용 공고(직무, 요구 기술 등)

- 출력 변수: 매칭 점수 (0~1의 값으로 매칭 정도를 표현)

알고리즘

- 추천 시스템: 협업 필터링, 콘텐츠 기반 필터링, 딥러닝 기반 추천 시스템
- 유사도 측정: 코사인 유사도, 유클리디안 거리, Jaccard 유사도
- 모델 평가 지표: Precision, Recall, F1-score, ROC AUC

구현 방법

- 유사도 계산: 구직자와 채용 공고 간의 유사도를 계산합니다. 예를 들어, 직무나 기술 스택에 대한 유사도를 측정하여 매칭 점수를 계산할 수 있습니다.
- 추천 알고리즘: 협업 필터링 방식이나 콘텐츠 기반 필터링 방식으로 구직자에게 적합한 직무를 추천합니다.

추천 도구

- **Python:** Scikit-learn, TensorFlow/Keras (추천 시스템 모델)
- 추천 시스템 라이브러리: Surprise, LightFM

4. 채용 시장 분석 및 트렌드 예측

목표

- 채용 공고 데이터를 분석하여 채용 시장의 트렌드(예: 인기 직무, 요구 기술 등)를 예측하고 분석합니다.

데이터

- 입력 변수: 채용 공고의 제목, 직무, 요구 기술 스택, 경력 수준, 지역 등
- 출력 변수: 채용 시장 트렌드(예: 특정 직무의 증가율, 기술 스택의 인기도 등)

알고리즘

- 시계열 분석: ARIMA, Prophet, LSTM (딥러닝)

- 분류/회귀 모델: 랜덤 포레스트, XGBoost, SVM
- 군집화: K-means, DBSCAN

구현 방법

- 시계열 분석: 특정 직무나 기술의 수요 변화를 예측합니다. 예를 들어, 특정 직무에 대한 채용 공고 수의 변화 추이를 분석하여 향후 수요를 예측할 수 있습니다.
- 군집화: 채용 공고를 군집화하여 주요 트렌드나 패턴을 발견합니다.

추천 도구

- **Python:** Pandas, Scikit-learn, TensorFlow/Keras, Prophet, ARIMA
 - 시계열 분석: Statsmodels, Facebook Prophet
-

5. 이상치 탐지 모델

목표

- 채용 공고 데이터에서 이상치(예: 잘못된 연봉, 비정상적인 경력 수준 등)를 탐지합니다.

데이터

- 입력 변수: 채용 공고의 연봉, 요구 경력, 학력 등
- 출력 변수: 이상치 여부 (이상치/정상)

알고리즘

- 이상치 탐지: Isolation Forest, One-Class SVM, DBSCAN, LOF (Local Outlier Factor)
- 모델 평가 지표: Precision, Recall, F1-score

구현 방법

- 특성 선택: 연봉, 경력 수준 등 중요한 특성을 기반으로 이상치를 탐지합니다.

- 이상치 탐지 모델: 이상치 탐지 알고리즘을 사용하여 데이터를 필터링하고, 이상치에 해당하는 항목을 분류합니다.

추천 도구

- **Python:** Scikit-learn, PyOD (이상치 탐지 라이브러리)