

Szkic analizy technologicznej

Anna Wolańska, Arkadiusz Jaciubek

Career Navigator

1. Wprowadzenie

Produkt opiera się na zaawansowanych technologiach Machine Learning (ML) i przetwarzania języka naturalnego (NLP), które razem tworzą system służący do analizy ofert pracy i dopasowania ich do profilu użytkownika. Całość infrastruktury jest planowana na chmurze AWS, co zapewnia skalowalność i elastyczność.

2. Architektura systemu

Moduły kluczowe:

- **Backend (Python) z integracją ML**
- **Frontend (React)**
- **Baza danych i API (PostgreSQL)**
- **Usługi NLP:** spaCy dla ekstrakcji i klasyfikacji nazw własnych
- **Model ML do klasyfikacji i dopasowania**

3. Technologie i komponenty

Komponent	Technologia	Opis
ML Model	TensorFlow / Keras	Model klasyfikacyjny do przewidywania roli zawodowej na podstawie umiejętności
Embedding	SentenceTransformer	Model do konwersji tekstu na wektory semantyczne (embeddingi)
NLP	spaCy	NER do identyfikacji nazw umiejętności, narzędzi, technologii

Backend	Python (FastAPI / Flask)	Serwer do obsługi API i integracji modeli ML
Frontend	React	Interfejs dla użytkownika, wyświetlanie dopasowanych ofert, rekomendacji
Chmura	AWS (EC2, S3, Lambda, RDS, SageMaker)	Hostowanie usług, skalowanie, przechowywanie danych i modeli

4. Analiza i przepływ danych

a. Przetwarzanie ofert pracy:

- Pobieranie danych o ofertach z różnych źródeł (np. API, pliki).
- Użycie spaCy do wyodrębniania kluczowych nazw (umiejętności, narzędzi).
- Reprezentacja opisów ofert za pomocą embeddingów (SentenceTransformer).

b. Profil kandydata:

- Użytkownik wprowadza deklarowane umiejętności.
- Umiejętności są konwertowane na embeddingi.
- Model klasyfikacyjny (keras) przewiduje najbardziej dopasowane role zawodowe.
- Analiza brakujących umiejętności w ofertach (np. za pomocą porównania embeddingów ofert i profilu).

c. Dopasowywanie ofert:

- Porównanie embeddingów ofert i profilu użytkownika (np. poprzez odległość cosinusową).
- Ranking ofert na podstawie podobieństwa i predykcji modelu.

d. Rekomendacje rozwojowe:

- Podpowiedzi brakujących umiejętności na podstawie różnicy embeddingów.

5. Model ML i jego funkcjonowanie

- **Model klasyfikacji** (zdefiniowany w module):
- **Działanie:**
 - Konwersja umiejętności na embeddingi.

- Przewidywanie roli za pomocą modelu Keras.
- Ranking najbardziej prawdopodobnych stanowisk.

6. Przechowywanie i zarządzanie danymi

- **Baza danych:** RDS (np. PostgreSQL/MySQL) do przechowywania danych ofert, profili użytkowników, wyników predykcji.
- **Przechowywanie modeli:** Amazon S3 lub SageMaker Model Registry.
- **Integracja API:** REST API do komunikacji między frontendem a backendem.