NLP Final Project Proposal

Project Title: Cross-Domain Job Matching: Transformer-based Semantic Retrieval between Job Descriptions and Resumes

Team Members: Cindy Gao, Cynthia Zhou, Jenny Wu

Project Objective: Design a tool that enhances traditional keyword-based resume screening by enabling semantic understanding between resumes and job descriptions. The project formulates a **cross-domain matching problem**: given a resume, identify the most semantically similar job descriptions and predict the **top three combinations of job field, title, and location** that best align with the candidate's background.

Datasets:

1. Job description: https://www.kaggle.com/datasets/PromptCloudHQ/us-jobs-on-monstercom/data
2. Resume: https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset

We will use the Monster.com Job Descriptions dataset as the training corpus and the Resume Dataset as the testing corpus. Job descriptions provide structured information about job title, field, and location after data preprocessing, while resumes contain unstructured skill sets and prior experience.

Model:

One Main model: Sentence-BERT

Step 1 – Keyword Extraction and Statistical Analysis

Apply TF-IDF/KeyBERT to extract representative keywords from each job description.
Aggregate the extracted keywords by job title, industry, and location to form a keyword–feature matrix highlighting the most salient skills for each domain.
Compute the industry opportunity distribution by counting job postings in each category and measuring keyword concentration (skill diversity and frequency).

- **Output 1:** Keyword-based feature repository linking skills to specific job titles, industries, and regions.
- *(Optional Output 2:)* Statistical summary of opportunity distribution across industries and locations.

Step 2 – Job Embedding Model Training

Utilize a Transformer-based encoder such as **Sentence-BERT (SBERT)** to obtain high-dimensional semantic embeddings for job descriptions.
Fine-tune the encoder on job-related data to capture domain-specific semantic relations.

Fine-tuning setup:

- *Positive pairs:* job descriptions with similar titles or industries.
- *Negative pairs:* job descriptions from distinct or unrelated fields.
- *Loss:* contrastive or triplet loss to minimize distance between semantically similar jobs while separating unrelated ones.

After fine-tuning, each job description is represented as a dense contextual embedding.
All vectors constitute the **Job Embedding Database**, serving as the base for semantic retrieval.

Step 3 – Evaluation and Storage

Evaluate embedding quality by measuring cosine similarity among semantically related job descriptions.
Store all vectors together with metadata (job title, field, location) in a **searchable index**—for instance, a FAISS or HNSW vector store—to enable efficient retrieval during matching.

Step 4 – Resume Embedding and Cross-Domain Matching

Clean and segment resumes into *Summary*, *Experience*, *Skills*, and *Education* sections.
Extract salient skill phrases or named entities (via TF-IDF/ KeyBERT) to build compact resume representations.
Encode each resume using the **same fine-tuned Transformer encoder** so that both resumes and job descriptions share a unified semantic space.
Normalize embeddings to enable cosine-similarity comparison.

Retrieval Procedure:

1. Compute cosine similarity between each resume embedding and all job embeddings.
2. Rank results by similarity scores.
3. Retrieve the **Top-3 most semantically relevant job matches**, each represented by *(Job Field, Title, Location)* with its similarity score and aligned keywords.

(For added complexity) — Add a lightweight **cross-encoder** re-ranking layer that jointly encodes each resume–job pair from the top-K retrieval results to refine ranking precision (e.g., using a smaller BERT-based cross-encoder fine-tuned on relevance labels).

Novelty:

1. Beyond traditional job-category classification, our model jointly embeds field and location as auxiliary semantic dimensions. This enables context-aware retrieval that captures how job titles vary across industries and regions. For example, “Data Analyst” roles in the Finance industry emphasize reporting, while in Healthcare focuses on patient data. The model outputs the top-3 most relevant (field + title + location) combinations for each candidate.
2. (For added complexity) Furthermore, our framework supports international candidates by handling multilingual resumes. Using either multilingual Transformer models or automatic translation into English, we enable cross-lingual matching between resumes and English job descriptions.