{  
 "cells": [  
 {  
 "cell\_type": "code",  
 "execution\_count": 2,  
 "metadata": {},  
 "outputs": [  
 {  
 "name": "stderr",  
 "output\_type": "stream",  
 "text": [  
 "2022-06-11 01:23:58.083 INFO numexpr.utils: NumExpr defaulting to 8 threads.\n"  
 ]  
 }  
 ],  
 "source": [  
 "from http.client import ImproperConnectionState\n",  
 "import streamlit as st\n",  
 "import pandas as pd\n",  
 "import numpy as np\n",  
 "from io import StringIO \n",  
 "from pyresparser import ResumeParser\n",  
 "import os \n",  
 "from docx import Document\n",  
 "from yaml import DocumentEndEvent\n",  
 "from PIL import Image\n",  
 "from utils import ngrams\n",  
 "from nltk.corpus import stopwords\n",  
 "from sklearn.metrics.pairwise import cosine\_similarity\n",  
 "from sklearn.feature\_extraction.text import TfidfVectorizer\n",  
 "from sklearn.neighbors import NearestNeighbors"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": 15,  
 "metadata": {},  
 "outputs": [  
 {  
 "data": {  
 "text/html": [  
 "<div>\n",  
 "<style scoped>\n",  
 " .dataframe tbody tr th:only-of-type {\n",  
 " vertical-align: middle;\n",  
 " }\n",  
 "\n",  
 " .dataframe tbody tr th {\n",  
 " vertical-align: top;\n",  
 " }\n",  
 "\n",  
 " .dataframe thead th {\n",  
 " text-align: right;\n",  
 " }\n",  
 "</style>\n",  
 "<table border=\"1\" class=\"dataframe\">\n",  
 " <thead>\n",  
 " <tr style=\"text-align: right;\">\n",  
 " <th></th>\n",  
 " <th>Unnamed: 0</th>\n",  
 " <th>url</th>\n",  
 " <th>Position</th>\n",  
 " <th>Company</th>\n",  
 " <th>Location</th>\n",  
 " <th>Job\_Description</th>\n",  
 " </tr>\n",  
 " </thead>\n",  
 " <tbody>\n",  
 " <tr>\n",  
 " <th>0</th>\n",  
 " <td>0</td>\n",  
 " <td>https://www.glassdoor.co.in/partner/jobListing...</td>\n",  
 " <td>Software Testing Internship</td>\n",  
 " <td>Smart Food Safe Solutions Inc</td>\n",  
 " <td>â€“ Bengaluru</td>\n",  
 " <td>About the company:\\nSmart Food Safe Solutions ...</td>\n",  
 " </tr>\n",  
 " <tr>\n",  
 " <th>1</th>\n",  
 " <td>1</td>\n",  
 " <td>https://www.glassdoor.co.in/partner/jobListing...</td>\n",  
 " <td>Embedded Software Testing</td>\n",  
 " <td>Mobiveil</td>\n",  
 " <td>â€“ Bengaluru</td>\n",  
 " <td>Location : Bangalore\\nExperience : 4+ Years\\n\\...</td>\n",  
 " </tr>\n",  
 " </tbody>\n",  
 "</table>\n",  
 "</div>"  
 ],  
 "text/plain": [  
 " Unnamed: 0 url \\\n",  
 "0 0 https://www.glassdoor.co.in/partner/jobListing... \n",  
 "1 1 https://www.glassdoor.co.in/partner/jobListing... \n",  
 "\n",  
 " Position Company Location \\\n",  
 "0 Software Testing Internship Smart Food Safe Solutions Inc â€“ Bengaluru \n",  
 "1 Embedded Software Testing Mobiveil â€“ Bengaluru \n",  
 "\n",  
 " Job\_Description \n",  
 "0 About the company:\\nSmart Food Safe Solutions ... \n",  
 "1 Location : Bangalore\\nExperience : 4+ Years\\n\\... "  
 ]  
 },  
 "execution\_count": 15,  
 "metadata": {},  
 "output\_type": "execute\_result"  
 }  
 ],  
 "source": [  
 "df =pd.read\_csv('job\_final.csv') \n",  
 "df.head(2)"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": 17,  
 "metadata": {},  
 "outputs": [],  
 "source": [  
 "stopw = set(stopwords.words('english'))"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": 18,  
 "metadata": {},  
 "outputs": [],  
 "source": [  
 "df['test']=df['Job\_Description'].apply(lambda x: ' '.join([word for word in str(x).split() if len(word)>2 and word not in (stopw)]))"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": 21,  
 "metadata": {},  
 "outputs": [  
 {  
 "data": {  
 "text/plain": [  
 "0 About company: Smart Food Safe Solutions Inc. ...\n",  
 "Name: test, dtype: object"  
 ]  
 },  
 "execution\_count": 21,  
 "metadata": {},  
 "output\_type": "execute\_result"  
 }  
 ],  
 "source": [  
 "df[\"test\"].head(1)"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": 24,  
 "metadata": {},  
 "outputs": [],  
 "source": [  
 "vectorizer = TfidfVectorizer(min\_df=1, analyzer=ngrams, lowercase=False)\n",  
 "def getNearestN(query):\n",  
 " queryTFIDF\_ = vectorizer.transform(query)\n",  
 " distances, indices = nbrs.kneighbors(queryTFIDF\_)\n",  
 " return distances, indices\n",  
 "\n",  
 "\n",  
 "cleaned\_skills = ['Sports Python Programming Javascript Flask Matplotlib Github Process Flower Html Postgresql System Tensorflow Sql Aws Docker Css International Pandas Api Ai Java']\n",  
 "tfidf = vectorizer.fit\_transform(cleaned\_skills)\n",  
 "nbrs = NearestNeighbors(n\_neighbors=1, n\_jobs=-1).fit(tfidf)\n",  
 "unique\_org = (df['test'].values)\n",  
 "distances, indices = getNearestN(unique\_org)\n",  
 "unique\_org = list(unique\_org)"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": 26,  
 "metadata": {},  
 "outputs": [  
 {  
 "data": {  
 "text/plain": [  
 "'Location Bangalore Experience Years Job Description Good different type testing proficient Python Based Test Script Automation Good experienced Embedded system testing module/component end end. Must willing test manually tests canâ€™t automated due complexity How Apply: Email resumes jobs@mobiveil.com job title subject line.'"  
 ]  
 },  
 "execution\_count": 26,  
 "metadata": {},  
 "output\_type": "execute\_result"  
 }  
 ],  
 "source": [  
 "unique\_org[1]"  
 ]  
 },  
 {  
 "cell\_type": "code",  
 "execution\_count": null,  
 "metadata": {},  
 "outputs": [],  
 "source": []  
 }  
 ],  
 "metadata": {  
 "kernelspec": {  
 "display\_name": "Python 3.9.12 ('base')",  
 "language": "python",  
 "name": "python3"  
 },  
 "language\_info": {  
 "codemirror\_mode": {  
 "name": "ipython",  
 "version": 3  
 },  
 "file\_extension": ".py",  
 "mimetype": "text/x-python",  
 "name": "python",  
 "nbconvert\_exporter": "python",  
 "pygments\_lexer": "ipython3",  
 "version": "3.9.12"  
 },  
 "orig\_nbformat": 4,  
 "vscode": {  
 "interpreter": {  
 "hash": "4af1e78d603f7d27cc60a9a5c8ae254cf839c35c66daa31c3a403b164618dcda"  
 }  
 }  
 },  
 "nbformat": 4,  
 "nbformat\_minor": 2  
}