```
from flask import Flask,render_template,url_for,request
import pandas as pd

import keras
from keras.preprocessing.text import one_hot, Tokenizer
from keras.preprocessing.sequence import pad_sequences
from keras.layers import Bidirectional,LSTM,Flatten,Dense,GRU,Conv1D,MaxPooling1D,Dropout
from keras.layers.embeddings import Embedding
from keras.models import Sequential
from keras.models import load_model
import numpy as np
from tensorflow.keras import layers
from tensorflow.keras import regularizers
from keras.regularizers import l2

from tensorflow.keras.models import load_model

#https://file.io/UUClmyYFpPG1
app = Flask(__name__)

@app.route('/')
def home():
 return render_template('home.html')

@app.route('/predict',methods=['POST'])

def predict():

    dataset= pd.read_csv('dataset.csv')
    X_train = dataset['text'].astype(str).values

    max_num_words = 70000
    max_length =50
    tokenizer = Tokenizer(num_words=max_num_words,filters='!"#$%&()*+,-./:;<=>?@[\\]^_`{|}~\t\n')

    tokenizer.fit_on_texts(X_train)


    model = load_model('models.h5',compile = False)
    if request.method == "POST":
        message= request.form['message']


        seq = tokenizer.texts_to_sequences([message])
        pad = pad_sequences(seq, maxlen = 50)
        pred = model.predict(pad)



        labels = ['English', 'Germain', 'French', 'Tamazigh']
        print(pred, labels[np.argmax(pred)])
        p = labels[np.argmax(pred)]

        return render_template('results.html',prediction = p, m = message)



if __name__ == '__main__':
 app.run(debug=True)
```