```
import streamlit as st
import pickle
import nltk
import string
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import CountVectorizer

def cleanup_text(message):
    message = message.translate(str.maketrans('','',string.punctuation))
    words = [stemmer.stem(w) for w in message.split() if w.lower() not in
stopwords.words('english') ]
    return ' '.join(words)


def load_model(path='spam_classifier.pkl'):
    with open(path,'rb') as f:
        return pickle.load(f)


base="light"
backgroundColor="#f3e1e1"

st.title('NATURAL LANGUAGE PROCESSING')

from PIL import Image
image = Image.open('nlp.jpg')
st.image(image, caption='NLP USING NLTK')

st.title('Email Spam detection')
with st.spinner('loading Spam classfication model'):
    model = load_model()
    vectorizer = load_model('count_vectorizer.pkl')

message = st.text_area('enter email subject for spam classification')
btn = st.button('click to process')
if btn and len(message)> 5:
    stemmer = PorterStemmer()
    clean_msg = cleanup_text(message)
    data = vectorizer.transform([clean_msg])
    data = data.toarray()
    prediction = model.predict(data)
    st.title('Prediction')
    if prediction[0] == 0:
        st.success("not spam Message")
    elif prediction[0] == 1:
        st.warning("Spam Message")
    else:
        st.error("try again")
```