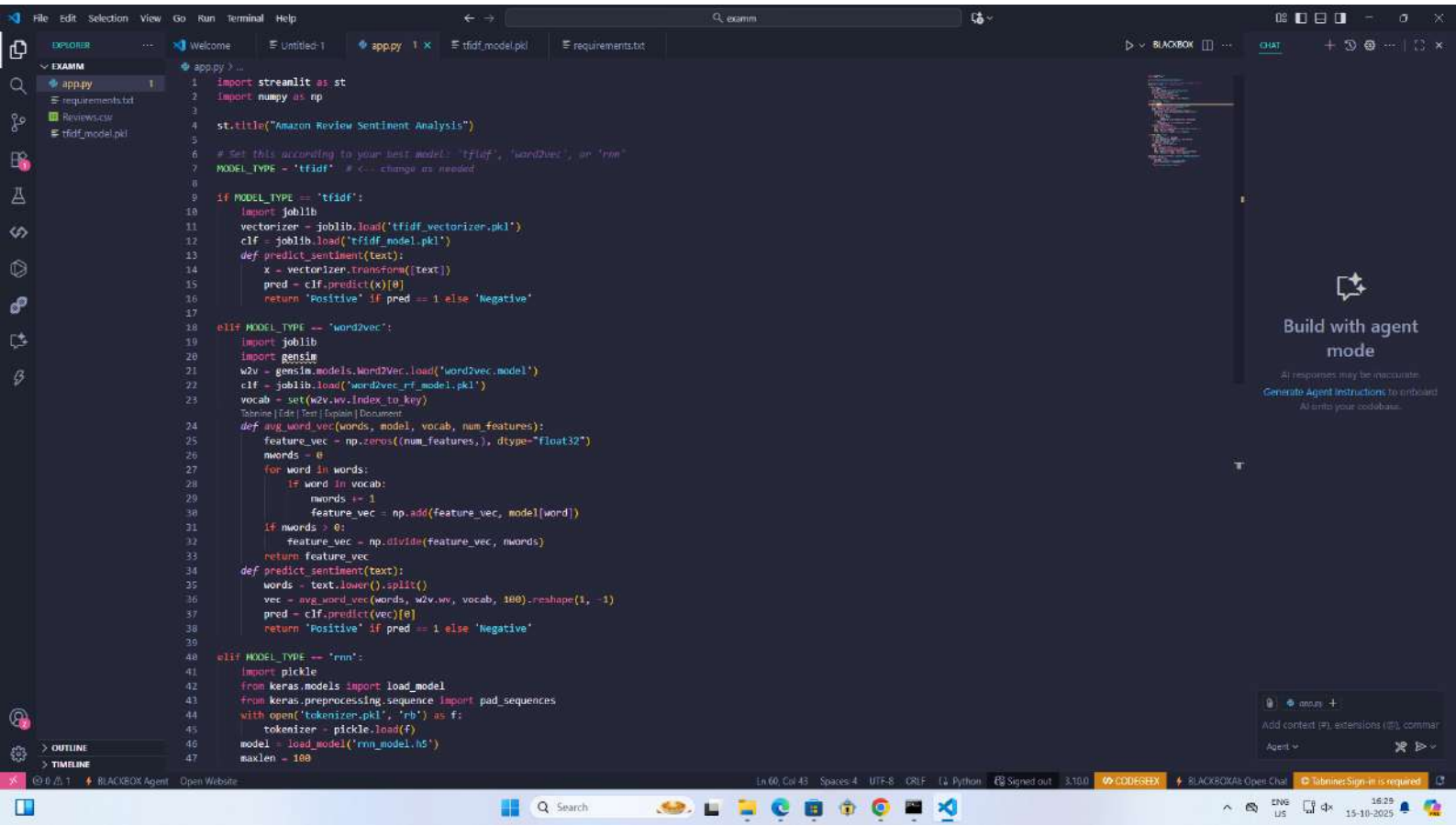
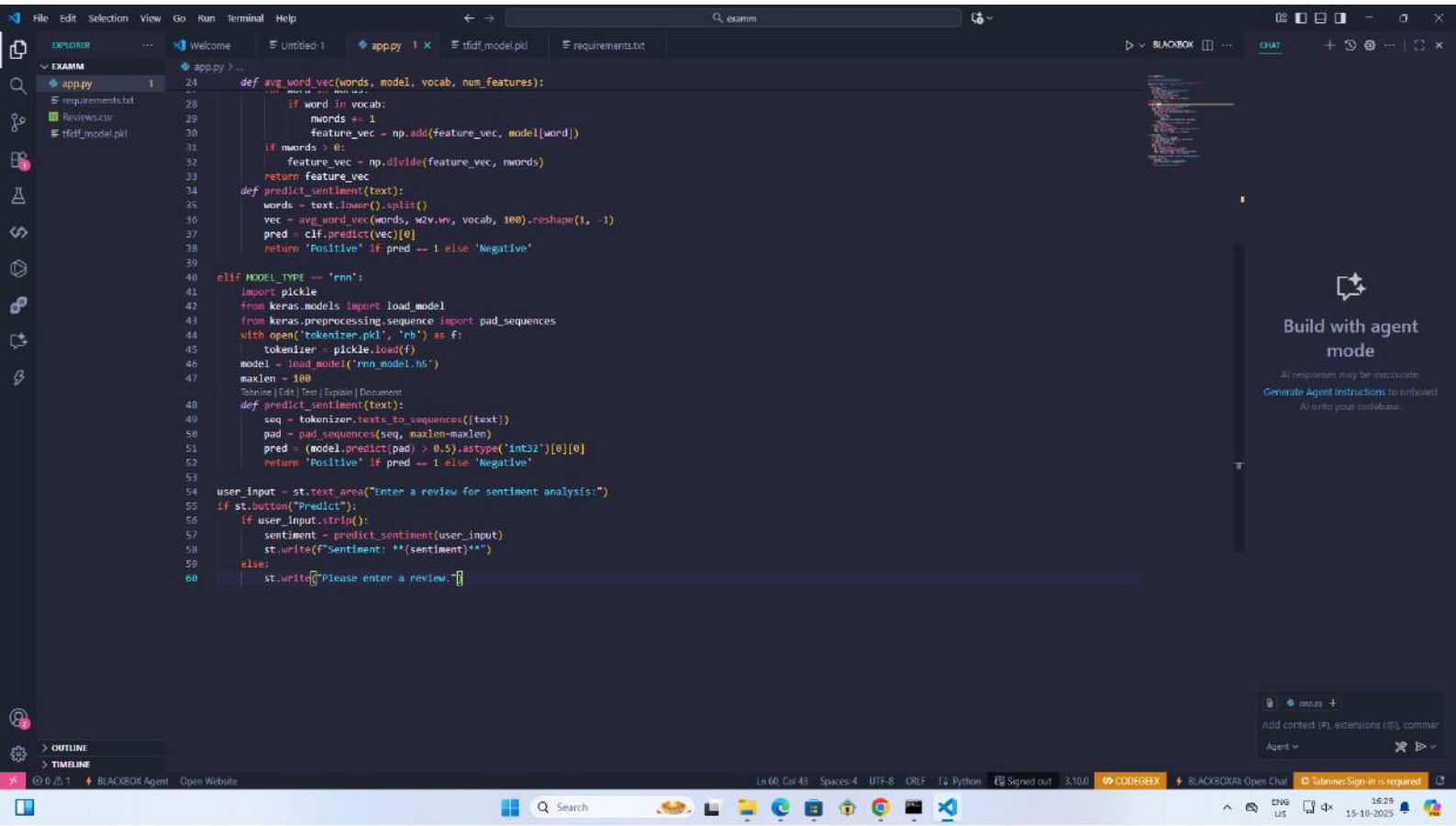
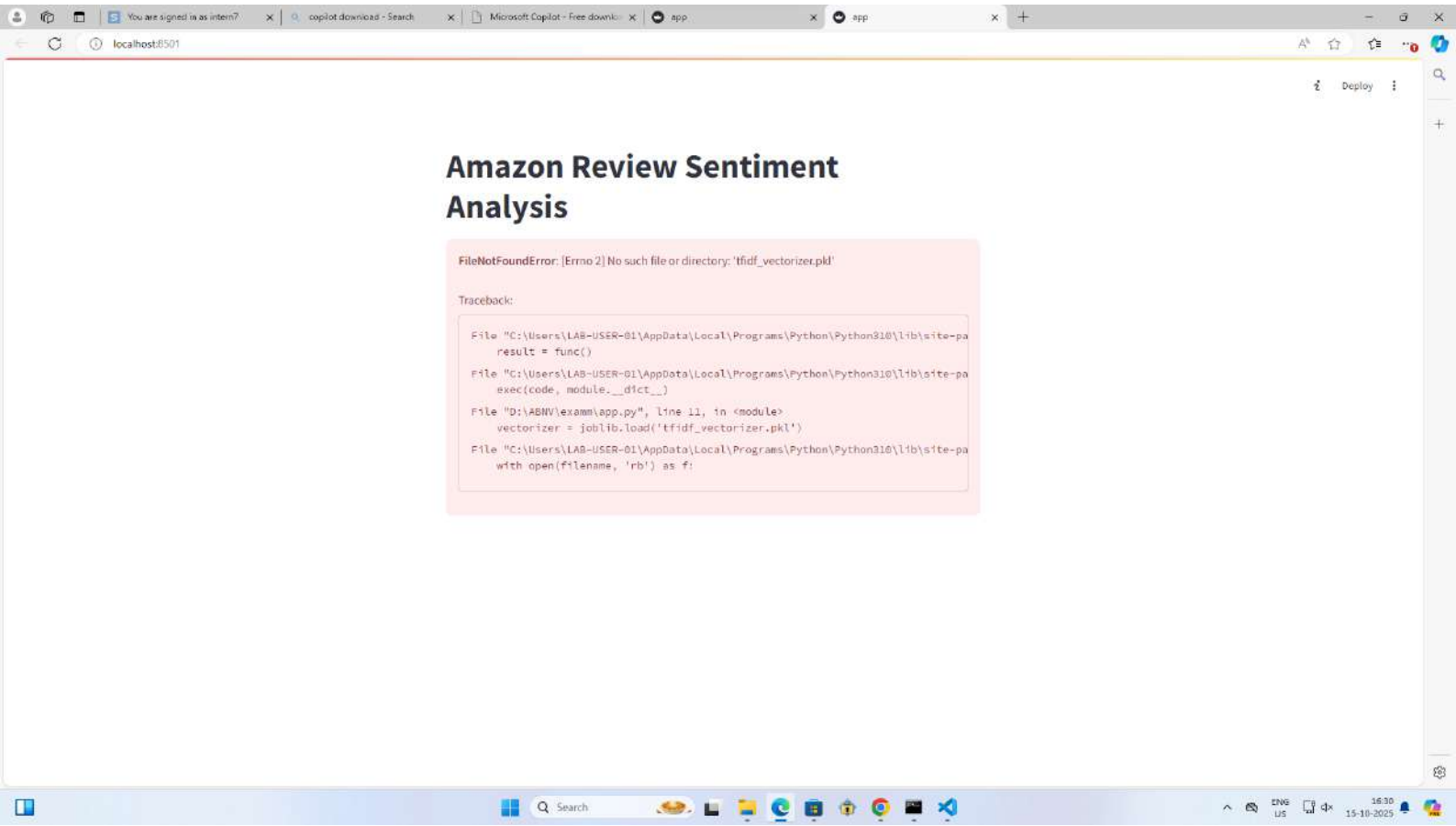

File Edit Selection View Go Run Terminal Help
examm
EXPLORER
EXAMM
app.py
requirements.txt
Reviews.csv
tfidf_model.pkl
Welcome
Untitled-1
app.py
tfidf_model.pkl
requirements.txt
BLACKBOX
CHAT
app.py > ...
import streamlit as st
import numpy as np
st.title("Amazon Review Sentiment Analysis")
# Set this according to your best model: 'tfidf', 'word2vec', or 'rnn'
MODEL_TYPE = 'tfidf' # <-- change as needed
if MODEL_TYPE == 'tfidf':
import joblib
vectorizer = joblib.load('tfidf_vectorizer.pkl')
clf = joblib.load('tfidf_model.pkl')
def predict_sentiment(text):
x = vectorizer.transform([text])
pred = clf.predict(x)[0]
return 'Positive' if pred == 1 else 'Negative'
elif MODEL_TYPE == 'word2vec':
import joblib
import gensim
w2v = gensim.models.Word2Vec.load('word2vec.model')
clf = joblib.load('word2vec_rf_model.pkl')
vocab = set(w2v.wv.index_to_key)
Tabnine | Edit | Test | Explain | Document
def avg_word_vec(words, model, vocab, num_features):
feature_vec = np.zeros((num_features,), dtype="float32")
nwords = 0
for word in words:
if word in vocab:
nwords += 1
feature_vec = np.add(feature_vec, model[word])
if nwords > 0:
feature_vec = np.divide(feature_vec, nwords)
return feature_vec
def predict_sentiment(text):
words = text.lower().split()
vec = avg_word_vec(words, w2v.wv, vocab, 100).reshape(1, -1)
pred = clf.predict(vec)[0]
return 'Positive' if pred == 1 else 'Negative'
elif MODEL_TYPE == 'rnn':
import pickle
from keras.models import load_model
from keras.preprocessing.sequence import pad_sequences
with open('tokenizer.pkl', 'rb') as f:
tokenizer = pickle.load(f)
model = load_model('rnn_model.h5')
maxlen = 100
Build with agent mode
AI responses may be inaccurate.
Generate Agent Instructions to onboard AI onto your codebase.
Add context (#), extensions (@), commands (/)
Agent
OUTLINE
TIMELINE
Ln 60, Col 43
Spaces: 4
UTF-8
CRLF
Python
Signed out
3.10.0
CODEGEEX
BLACKBOXAI: Open Chat
Tabnine: Sign-in is required
Search
ENG
US
16:29
15-10-2025


File Edit Selection View Go Run Terminal Help
examm
EXPLORER
EXAMM
app.py
requirements.txt
Reviews.csv
tfidf_model.pkl
Welcome
Untitled-1
app.py
tfidf_model.pkl
requirements.txt
BLACKBOX
CHAT
app.py > ...
def avg_word_vec(words, model, vocab, num_features):
if word in vocab:
nwords += 1
feature_vec = np.add(feature_vec, model[word])
if nwords > 0:
feature_vec = np.divide(feature_vec, nwords)
return feature_vec
def predict_sentiment(text):
words = text.lower().split()
vec = avg_word_vec(words, w2v.wv, vocab, 100).reshape(1, -1)
pred = clf.predict(vec)[0]
return 'Positive' if pred == 1 else 'Negative'
elif MODEL_TYPE == 'rnn':
import pickle
from keras.models import load_model
from keras.preprocessing.sequence import pad_sequences
with open('tokenizer.pkl', 'rb') as f:
tokenizer = pickle.load(f)
model = load_model('rnn_model.h5')
maxlen = 100
Tabnine | Edit | Test | Explain | Document
def predict_sentiment(text):
seq = tokenizer.texts_to_sequences([text])
pad = pad_sequences(seq, maxlen=maxlen)
pred = (model.predict(pad) > 0.5).astype('int32')[0][0]
return 'Positive' if pred == 1 else 'Negative'
user_input = st.text_area("Enter a review for sentiment analysis:")
if st.button("Predict"):
if user_input.strip():
sentiment = predict_sentiment(user_input)
st.write(f"Sentiment: **{sentiment}**")
else:
st.write("Please enter a review.")
Build with agent mode
AI responses may be inaccurate.
Generate Agent Instructions to onboard AI onto your codebase.
Add context (#), extensions (@), commands (/)
Agent
OUTLINE
TIMELINE
BLACKBOX Agent
Open Website
Ln 60, Col 43
Spaces: 4
UTF-8
CRLF
Python
Signed out
3.10.0
CODEGEEX
BLACKBOX AI: Open Chat
Tabnine: Sign-in is required
Search
ENG
US
16:29
15-10-2025


You are signed in as intern7
copilot download - Search
Microsoft Copilot - Free downlo
app
app
localhost:8501
Deploy
Amazon Review Sentiment Analysis
FileNotFoundError: [Errno 2] No such file or directory: 'tfidf_vectorizer.pkl'
Traceback:
File "C:\Users\LAB-USER-01\AppData\Local\Programs\Python\Python310\lib\site-pa
result = func()
File "C:\Users\LAB-USER-01\AppData\Local\Programs\Python\Python310\lib\site-pa
exec(code, module.__dict__)
File "D:\ABNV\examm\app.py", line 11, in <module>
vectorizer = joblib.load('tfidf_vectorizer.pkl')
File "C:\Users\LAB-USER-01\AppData\Local\Programs\Python\Python310\lib\site-pa
with open(filename, 'rb') as f:
Search
ENG
US
16:30
15-10-2025