```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Download NLTK resources
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')

def preprocess_text(text):
    # Tokenization
    words = word_tokenize(text.lower())

    # Remove stopwords
    stop_words = set(stopwords.words('english'))
    filtered_words = [word for word in words if word not in stop_words]

    # Lemmatization
    lemmatizer = WordNetLemmatizer()
    lemmatized_words = [lemmatizer.lemmatize(word) for word in filtered_words]

    return ' '.join(lemmatized_words)

def calculate_cosine_similarity(text1, text2):
    # Preprocess the texts
    processed_text1 = preprocess_text(text1)
    processed_text2 = preprocess_text(text2)

    # Calculate TF-IDF vectors
    tfidf_vectorizer = TfidfVectorizer()
    tfidf_matrix = tfidf_vectorizer.fit_transform([processed_text1, processed_text2])

    # Calculate cosine similarity between the two documents
    cosine_sim = cosine_similarity(tfidf_matrix[0], tfidf_matrix[1])[0][0]

    return cosine_sim

def detect_plagiarism(text1, text2, threshold=0.8):
    similarity_score = calculate_cosine_similarity(text1, text2)
    if similarity_score >= threshold:
        print("Plagiarism detected!")
        print("Similarity Score:", similarity_score)
    else:
        print("No plagiarism detected.")
        print("Similarity Score:", similarity_score)

# Test the plagiarism detection function
text1 = "This is an original text."
text2 = "This is a slightly modified version of the original text."
detect_plagiarism(text1, text2)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
[nltk_data] Downloading package wordnet to /root/nltk_data...
No plagiarism detected.
Similarity Score: 0.5023287782256718
```