from nltk.util import ngrams

from sklearn.metrics import jaccard\_score

import numpy as np

def get\_ngrams(text, n=2):

"""Generate n-grams from a string"""

return set([''.join(gram) for gram in ngrams(text, n)])

def jaccard\_similarity(text1, text2, n=2):

"""Calculate Jaccard Similarity using n-grams"""

ngrams1 = get\_ngrams(text1, n)

ngrams2 = get\_ngrams(text2, n)

intersection = len(ngrams1.intersection(ngrams2))

union = len(ngrams1.union(ngrams2))

return intersection / union if union != 0 else 0

# Example Usage

text1 = "hello"

text2 = "hallo"

similarity = jaccard\_similarity(text1, text2, n=2)

print(f"Jaccard Similarity (Bigram) between '{text1}' and '{text2}': {similarity:.2f}")

**Explanation:**

* **Input Strings:**
  + "hello" → **Bigrams**: { "he", "el", "ll", "lo" }
  + "hallo" → **Bigrams**: { "ha", "al", "ll", "lo" }
* **Intersection**: { "ll", "lo" } → **2 elements**
* **Union**: { "he", "el", "ll", "lo", "ha", "al" } → **6 elements**
* **Jaccard Similarity Calculation**: J=26=0.33J = \frac{2}{6} = 0.33J=62​=0.33