import difflib

import itertools

def ngrams(string, n=2):

    # Convert input string to lower-case and remove non-alphanumeric characters

    string = ''.join(e for e in string.lower() if e.isalnum())

    # Generate n-grams

    ngrams = zip(\*[string[i:] for i in range(n)])

    return [''.join(ngram) for ngram in ngrams]

def fuzzy\_ngram\_match(word1, word2, n=2):

    # Generate n-grams for each word

    ngrams1 = ngrams(word1, n)

    ngrams2 = ngrams(word2, n)

    # Use difflib to calculate the similarity between n-gram sets

    similarity = difflib.SequenceMatcher(None, ngrams1, ngrams2).ratio()

    return similarity

# Example usage:

word1 = "123$"

word2 = "123.5"

match\_score = fuzzy\_ngram\_match(word1, word2, n=2)

print(f"The similarity score between '{word1}' and '{word2}' is: {match\_score:.2f}")

text\_source = widgets.Text(value='srinivas', description='Source:', placeholder='Type something')

text\_destination = widgets.Text(value='sr in iva', description='Destination:', placeholder='Type something')

button\_predict = widgets.Button(description='Predict')

output = widgets.Output()

def on\_button\_clicked(b):

    with output:

        output.clear\_output(wait=True)

        word1 = text\_source.value

        word2 = text\_destination.value

        df\_input = prepare\_features(word1, word2)

        prediction = model.transform(df\_input).select("prediction").collect()[0][0]

        word1 = text\_source.value.strip()

        word2 = text\_destination.value.strip()

        ngram\_score = fuzzy\_ngram\_match(word1, word2)

        print(f'Similarity Score: {ngram\_score:.2f}')

        print("Please enter valid alphanumeric strings.")

        labels = {

                0: 'Case Sensitivity',

                1: 'Extra Space Issues',

                2: 'Special Character Differences',

                3: 'Rounded Off Numbers',

                4: 'Currency Symbol Difference',

                5: 'Leading Zero Issue',

                6: 'Negative vs Positive',

                7: 'Scientific Notation Difference',

                8: 'Thousands Separator Difference',

                9: 'No Match'

            }

        print(f'Prediction: {labels[int(prediction)]}')

button\_predict.on\_click(on\_button\_clicked)

display(widgets.VBox([text\_source, text\_destination, button\_predict, output]))