String Distance

One of the simplest ways to compare an LLM or chain's string output against a reference label is by using string distance measurements such as Levenshtein or postfix distance. This can be used alongside approximate/fuzzy matching criteria for very basic unit testing.

This can be accessed using the `string_distance` evaluator, which uses distance metric's from the rapidfuzz library.

Note: The returned scores are *distances*, meaning lower is typically "better".

For more information, check out the reference docs for the StringDistanceEvalChain for more info.

```
# %pip install rapidfuzz
```

```
from langchain.evaluation import load_evaluator

evaluator = load_evaluator("string_distance")
```

API Reference:

- load_evaluator from `langchain.evaluation`

```
evaluator.evaluate_strings(
    prediction="The job is completely done.",
    reference="The job is done",
)
```

```
    {'score': 0.11555555555555552}
```

```
# The results purely character-based, so it's less useful when negation
is concerned
evaluator.evaluate_strings(
    prediction="The job is done.",
    reference="The job isn't done",
)
```

```
    {'score': 0.0724999999999999}
```

Configure the String Distance Metric

By default, the `StringDistanceEvalChain` uses levenshtein distance, but it also supports other string distance algorithms. Configure using the `distance` argument.

```
from langchain.evaluation import StringDistance

list(StringDistance)
```

API Reference:

- StringDistance from `langchain.evaluation`

```
    [<StringDistance.DAMERAU_LEVENSHTEIN: 'damerau_levenshtein'>,
     <StringDistance.LEVENSHTEIN: 'levenshtein'>,
     <StringDistance.JARO: 'jaro'>,
     <StringDistance.JARO_WINKLER: 'jaro_winkler'>]
```

```
jaro_evaluator = load_evaluator(
    "string_distance", distance=StringDistance.JARO
)
```

```
jaro_evaluator.evaluate_strings(
    prediction="The job is completely done.",
    reference="The job is done",
)
```

```
    {'score': 0.19259259259259254}
```

```
jaro_evaluator.evaluate_strings(
    prediction="The job is done.",
    reference="The job isn't done",
)
```

```
    {'score': 0.12083333333333324}
```