Approximate text matching with the `stringdist` package

Mark van der Loo

Statistics Netherlands

*useR!*2014

markvanderloo.eu

@MarkPJvanderLoo

The `stringdist` package

Fuzzy dictionary lookup

`amatch`	Fuzzy matching equivalent of `match`
`ain`	Fuzzy matching equivalent of `%in%`

The stringdist package

Fuzzy dictionary lookup

`amatch`	Fuzzy matching equivalent of `match`
`ain`	Fuzzy matching equivalent of `%in%`

String metrics

`stringdist`	Pairwise distances
`stringdistmatrix`	Distance matrix
`qgrams`	Compute *q*-gram profile

The stringdist package

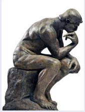
Fuzzy dictionary lookup

`amatch`	Fuzzy matching equivalent of `match`
`ain`	Fuzzy matching equivalent of `%in%`

String metrics

`stringdist`	Pairwise distances
`stringdistmatrix`	Distance matrix
`qgrams`	Compute q-gram profile

Design “philosophy”



Create interfaces that resemble base R (e.g. `match`, `adist`, `nchar`, `agrep`)

Dictionary lookup

```
> match("leia", c("leela","leia"))
[1] 2
```

Dictionary lookup

```
> match("leia", c("leela","leia"))
[1] 2
> match("liea", c("leela","leia"))
[1] NA
```

Dictionary lookup

```
> match("leia", c("leela","leia"))
[1] 2
> match("liea", c("leela","leia"))
[1] NA
> amatch("liea", c("leela","leia"), maxDist=1)
[1] 2
```

Dictionary lookup

```
> match("leia", c("leela","leia"))
[1] 2
> match("liea", c("leela","leia"))
[1] NA
> amatch("liea", c("leela","leia"), maxDist=1)
[1] 2
> "liea" %in% c("leela","leia")
[1] FALSE
```

Dictionary lookup

```
> match("leia", c("leela","leia"))
[1] 2
> match("liea", c("leela","leia"))
[1] NA
> amatch("liea", c("leela","leia"), maxDist=1)
[1] 2
> "liea" %in% c("leela","leia")
[1] FALSE
> ain("liea", c("leela","leia"), maxDist=1)
[1] TRUE
```

String distance

String distances

Implemented in the package

- edit-based distances
- *q*-gram based distances
- heuristic distances

Review papers

- L. Boytsov (2011). ACM Journal of Experimental Algorithmics **16** 1–86.
- G. Navarro (2001). ACM Computing Surveys **33** 31–88.

`stringdist` paper

- M.P.J. van der Loo (2014). *The* `stringdist` *package for approximate string matching.* The R Journal **6** *xx-xx*.

Edit-based distances

Definition

Count the minimum number of (weighted) basic operations that turns string s into string t.

Edit-based distances

Definition

Count the minimum number of (weighted) basic operations that turns string s into string t.

	Allowed operation			
Distance	substitution	deletion	insertion	transposition
Hamming	✔	✖	✖	✖
LCS	✖	✔	✔	✖
Levenshtein	✔	✔	✔	✖
OSA	✔	✔	✔	✔*
Damerau-Levenshtein	✔	✔	✔	✔

*Substrings may be edited only once.

Edit-based distances

Definition

Count the minimum number of (weighted) basic operations that turns string s into string t.

	Allowed operation			
Distance	substitution	deletion	insertion	transposition
Hamming	✔	✖	✖	✖
LCS	✖	✔	✔	✖
Levenshtein	✔	✔	✔	✖
OSA	✔	✔	✔	✔*
Damerau-Levenshtein	✔	✔	✔	✔

*Substrings may be edited only once.

```
> stringdist('leia','liea',method='hamming')

[1] 2
```

Edit-based distances

Definition

Count the minimum number of (weighted) basic operations that turns string s into string t.

	Allowed operation			
Distance	substitution	deletion	insertion	transposition
Hamming	✔	✖	✖	✖
LCS	✖	✔	✔	✖
Levenshtein	✔	✔	✔	✖
OSA	✔	✔	✔	✔*
Damerau-Levenshtein	✔	✔	✔	✔

*Substrings may be edited only once.

```
> stringdist('leia','liea',method='hamming')
[1] 2
> stringdist('leia','liea',method='dl')
[1] 1
```

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

banana

Q:

$\mathbf{x}$:

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

$\boxed{\text{ba}}\text{nana}$

Q:	ba
$\mathbf{x}$:	1

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

b$\boxed{\text{an}}$ana

Q:	ba	an
$\mathbf{x}$:	1	1

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

ba$\boxed{\text{na}}$na

Q:	ba	an	na
$\mathbf{x}$:	1	1	1

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

ban$\boxed{\text{an}}$a

Q:	ba	an	na
$\mathbf{x}$:	1	2	1

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

bana$\boxed{\text{na}}$

Q:	ba	an	na
$\mathbf{x}$:	1	2	2

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

Jaccard $\dfrac{|Q_1 \cap Q_2|}{|Q_1 \cup Q_2|}$

```
> stringdist('leia','leela'
+   , method='jaccard',q=2)
[1] 0.8333333
```

Cosine $\cos\left(\mathbf{x}\angle\mathbf{y}\right)$

```
> stringdist('leia','leela'
+   , method='cosine',q=2)
[1] 0.7113249
```

q-gram based distances

Definition

Any (vector) distance between two q-gram profiles.

```
> qgrams(x = 'leia',y = 'leela',q=2)

  le ei ia la el ee
x  1  1  1  0  0  0
y  1  0  0  1  1  1
```

Heuristic distances: Jaro-Winkler

Definition

- It's complicated :-).
- Intended for human-typed name/address data

```
> stringdist('liea','leia',method='jw',p=0.1)
```

```
[1] 0.075
```

- Ranges from 0 (equal) to 1 (dissimilar).
- $0 \leq p \leq 0.25$: emphasis on first 4 characters.
- $p = 0$: Jaro-distance

Character encoding

Image from `https://code.google.com/p/tworsekey/`

Character encoding

```
> stringdist(’ö’,’o’)
[1] 1 # Replace one symbol

> stringdist(’ö’,’o’,useBytes=TRUE)
[1] 2 # delete one byte, replace another (utf-8)
```

Missing values

WHOLE MILK
Milk
MISSING
Name : Value Six
Missing Since : 30-06-2014
Last seen at : input.csv
If you have seen Value Six or know someone who has, please contact your local statistician at +31 415 926 53

Handling missing values

```
> NA == NA
```

Handling missing values

```
> NA == NA
[1] NA
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
> match(NA, NA)
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
> match(NA, NA)
[1] 1 # <- note the useR's OMGWTFBBQ right there
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
> match(NA, NA)
[1] 1 # <- note the useR's OMGWTFBBQ right there
> amatch(NA, NA)
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
> match(NA, NA)
[1] 1 # <- note the useR's OMGWTFBBQ right there
> amatch(NA, NA)
[1] 1 # <- ok, at least we're consistent
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
> match(NA, NA)
[1] 1 # <- note the useR's OMGWTFBBQ right there
> amatch(NA, NA)
[1] 1 # <- ok, at least we're consistent
> amatch(NA, NA, matchNA=FALSE)
```

Handling missing values

```
> NA == NA
[1] NA
> adist(NA, NA)
[1] NA
> stringdist(NA, NA)
[1] NA
> match(NA, NA)
[1] 1 # <- note the useR's OMGWTFBBQ right there
> amatch(NA, NA)
[1] 1 # <- ok, at least we're consistent
> amatch(NA, NA, matchNA=FALSE)
[1] NA
```

Parallelization

For a single call:

```
> stringdistmatrix(a,b,ncores=4)
```

Or, define your own cluster:

```
> cl <- makeCluster(4)
> stringdistmatrix(a, b, cluster=cl)
> stringdistmatrix(c, d, cluster=cl)
> stopCluster(cl)
```

Performance

- stringdist(method='lv')
- About 30% faster than adist
- About 2 times faster then RecordLinkage

When comparing strings of 5 - 25 characters

Summary

- Nine different string metrics; core in `C99` (sorry Dirk :-))
- Approximate dictionary lookup
- Proper handling of encoding and missing values
- Fast
- Paralellization built in

Thank you for your attention!

t : `@markpjvanderloo`

e : `mark.vanderloo@gmail.com`

w : `markvanderloo.eu`