Approximate String Matching

Jeremy Nicholson and Justin Zobel and Karin Verspoor

Semest

THE UNIVERSITY OF MELBOURNE

Summary

Week 3:

- Approximate String Search and Matching
-
-
  - Edit Distance
  - N-Gram Distance
  - [Phonetic methods]
- Evaluation
- [Genomics]

Exact String Search

Consider:

-
-


```
In exes for foxes rex dux mixes a pox of waxed luxe
An axe, and an axon, to exo Exxon max oxen.
Grexit or Brexit as quixotic laxxers with b
```


Not (really) a Knowledge Technology!

Approximate String Search

Find `exon` in:

Not present!

...But what is the "closest" or "best" match?

This is a Knowledge Technology!

Important problems

Spelling Correction

Spelling Correction

- Need the notion of a **dictionary**:
  -
  -
  -
    misspelled
  - An item in the input which *does* correctly spelled *or* misspelled (
    of this subject)

Spelling Correction

Other Problems of Interest

- Computational Genomics (later, if we have time)
-
-
-
- Data cleaning
- ...

What's a "best" match?

Neighbourhood Search

For a given string *w* of interest:

- s
- **All** results found in dictionary are returned

Unix command-line utility agrep is an example.

Neighbourhood Search

Requires 1 insertion (o) so intended wo
neighbourhood search (and some uninte

Neighbourhood Search Efficiency

- Neighbourhood search is surprisingly fast!

- ...But Σ is a small constant, string of interest is usually short, and k is usually small
- For each neighbour, need a dictionary read (
- Binary search yields $\mathcal{O}(|w|^k \log D)$

Neighbourhood Search Effectiveness

Global Edit Distance

Global Edit Distance

Each operation is associated with a score;
Best match is the dictionary entry with best ag

Global Edit Distance

For example:

Item of interest: `crat`

`crat` → `cart`:

Match `c` (+1), Delete `r` (-1), Match `a` +1) = +1

`crat` → `arts`:

Replace `c` with `a` (-1), Match `r` (+1), sert `s`

(-1) = -1

`cart` is the better match

Global Edit Distance Parameter

This is the Levenshtein Distance (which is a “
number of edits required to transform one str

Global Edit Distance Parameter

- Hypothetically, any parameter is possible!

aba

- `foo`: Insert, Delete, Insert, Delete, Ins
- `aba`: Match, Match, Match = +12
- `cbc`: Replace, Match, Replace = +4

Global Edit Distance Parameter

Consider:

Is `faxing` more likely to be `facing`

Global Edit Distance Algorithm

From string f to string t, given array A of $|f|+1$ columns and $|t|+1$ rows, we can solve using the Needleman–Wunsch algorithm:

```
lf = strlen(f); lt = strlen(t);
A[0][0]=0;

for (k=1; k<=lf; k++)
    A[j][k] = max3( //Or min3 if m<i,d,r
        A[j][k-1] + d, //Deletion
        A[j-1][k] + i, //Insertion
        A[j-1][k-1] + equal(f[k-1],t[j-1])); //Replace or match
```

`equal()` returns m if characters match, r otherwise

Final score is at $A[lt][lf]$

Global Edit Distance in Action

In action: from `cat` to `arts`, Match (+1), Insert/Delete/Replace (-1)

					r
t	-3	-3	-1	-1	0
s	-4	-4	-2	-2	-1

Global Edit Distance: -1 (Replace, Match, Delete, Match, Insert)

More parameter concerns

Algorithm actually depends on parameter!


```
A[j-1][k-1] + equal(f[k-1],t[j-1]); //Replace or match
```

→ Match score greater than Insert/Del

e.g. Match (+1), Insert/Delete/Replace (-

More parameter concerns

- Algorithm actually depends on parameter!

→ Match score less than Insert/Delete

e.g. Match (0), Insert/Delete/Replace (+1)

(Levenshtein Distance)

Local Edit Distance

lengths, e.g. a word and a sentence

Local Edit Distance Algorithm

From string f to string t, given array A of $|f| + 1$ columns and $|t| + 1$ rows, we can solve using the Smith–Waterman algorithm:


```
lf = strlen(f); lt = strlen(t);
A[0][0]=0;
```



```
for (k=1; k<=lf; k++)
   A[j][k] = max4( //Or min4 if m<i,d,r
      0,
      A[j][k-1] + d, //Deletion
      A[j-1][k] + i, //Insertion
      A[j-1][k-1] + equal(f[k-1],t[j-1])); //Replace or match
```


`equal()` returns m if characters match, r otherwise

Final score is greatest value in the entire table (or least value, if $m < i, d, r$)

Local Edit Distance in Action

In action: from `cart` to `arts`; Match (+1), Insert/Delete/Replace (-1)

		r			
t	0	0	0	1	3
s	0	0	0	0	2

Best match: `art` with `art` (+3); ties are possible.

Edit Distance Efficiency

For strings f and t, Both algorithms above are $\mathcal{O}(|f||t|)$ in both space and time. (Space can be improved, but time (probably) cannot.)

hich we want

$\mathcal{O}(|f| \sum_{t \in D} |t|)$

Hence, integer comparisons are roughly t
dictionary. Whether this is feasible depen

N-Gram Distance

N-Gram Distance has same goal as Edit Distance: compare two strings to determine "best" match

2-grams of `cart`: `#c`, `ca`, `ar`, `rt`, `t#`
2-grams of `arts`: `#a`, `ar`, `rt`, `ts`, `s#`

N-Gram Distance between n-grams (s) and (t):
$|G_n(s)| + |G_n(t)| - 2 \times |G_n(s) \cap G_n$

N-Gram Distance

n-gram: substring of length n

$|G_2(\texttt{crat})| + |G_2(\texttt{cart})| - 2 \times |G_2($
$= 5 + 5 - 2 \times 2 = 6$ (better)

2-Gram Distance between crat and

$|G_2(\texttt{crat})| + |G_2(\texttt{arts})| - 2 \times |G_2($
$= 5 + 5 - 2 \times 0 = 10$

N-Gram Distance Efficiency

- Occasionally useful as a simpler variant of Edit Distance

- Despite its simplicity, takes roughly the same time to compare entire dictionary
- Quite useless for very long strings and/or ve

Orthography

In English (and some other languages), **orthography** (spelling) isn't a

`George O'Connell`

Also relevant in spelling correction (Englis
correctly!)

Soundex

One mechanism: Soundex

aeiouy	$\rightarrow$	0 (vowels)
bpfv	$\rightarrow$	1 (labials)
cgjkqsxz		2 (misc: fricatives, velars, etc.)

Four step process:

1. Except for initial character, translate table
2. Remove duplicates (e.g. $4444 \rightarrow 4$)
3. Remove 0s
4. Truncate to four symbols

Soundex

One mechanism: Soundex

`aehiouwy`	$\rightarrow$	0 (vowels)
`bpfv`		1 (labials)

tc.)

`r` $\rightarrow$ 6 (rhotic)

Four step process:

`king`	`kyngge`
`k052`	`k05220`
`k052`	`k0520`
`k52`	`k52`

Other Phonetic Methods

Better phonetic methods make use of the fact that some letters sounds

lpadist uses a text–to–sound algorithm t
the International Phonetic Alphabet (but c

There are also worse variants, like Phonix.

Evaluating an Approximate Matching System

Evaluation: consider whether the system is effective at solving the user's problem

To evaluate, we need:

- A number of cases of misspelled word
- The intended (correct) word for each
- An **evaluation metric**

Evaluation Metrics for Spelling Correction

We have some cases:

			ght/Wrong?
cracheyt	crotchety	cachet	✓
...	...		

Accuracy: fraction of correct responses (–

Evaluation Metrics for Spelling Correction

More realistic situation:

Misspelled Word	Correct Word	Predicted Word	Right/Wrong?
			×
			✓
			×
			✓
		carrier	
cracheyt	crotchety		
...	...		

Precision: fraction of correct responses among attempted responses ($\frac{2}{5}$)

Recall: proportion of words with a correct response (somewhere) ($\frac{2}{3}$)

Comparing Systems

- Typically, the value of the evaluation metric has little intrinsic meaning
  - "This system gets 81% accuracy" — useful for users, or not?

  - the accuracy becomes 74%"
  - "System A gets 45% precision and 80% reca
    System B gets 95% precision and 10% recal
    — Which one should we use? (Also: why?)

The answer depends on the problem (and the user)!

Summary

- What is approximate string search?
- ch;
- string? What do we need to generate them?
- How can we evaluate a typical approxi

Background Readings

Needleman, Saul B. and Wunsch, Christian D. (1970). “A general method applicable to the search for similarities in the amino acid sequence of two proteins”. Journal of Molecular Biology 48 (3): 44353. doi:10.1016/0022-2836(70)90057-4

(Originally in Russian, published in English as:) Levenshtein, Vladimir I.

47:

195197. doi:10.1016/0022-2836(81)9

Kondrak, Grzegorz (2005). “N-Gram Sim
Proceedings of the 12th international conf
and Information Retrieval (SPIRE’05), pp. 115-126, Buenos Aires, Argentina.

Zobel, Justin and Dart, Philip (1996). “Phonetic String Matching: Lessons from Information Retrieval”. In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR’96), pp. 166-172, New York, USA.

Extension Readings

Whitelaw, Casey and Hutchison, Ben and Chung, Grace Y and Ellis, Gerard (2009). “Using the Web for Language Independent

Ahmad, Farooq and Kondrak, Grzegorz (2005). “Learning a Spelling Error Model from Search Query Logs”. In Pr
Technology Conference and Conferen
Natural Language Processing (HLT/EM
Vancouver, Canada.

g)

- Possibly with “errors” (nucleotide/a

Typical Genomics problem:

- r
-
- But **much** larger strings: a small gen comparing perhaps 1K character se character sequences; alphabet is s

Suitability for Computational Genomics

Recall: we have a “short” (1K character) nucleotide/amino acid

sequence of interest, they might be susceptible to some medical condition

We’re allowed ∼ 16 errors; alphabet is

Suitability for Computational Genomics

Suitability for Computational Genomics

→ Prefers shorter chromosomes (not int

Suitability for Computational Genomics

... Seems like the right idea.

Suitability for Computational Genomics

... Can't fit table into memory.

... Requires approximate solutions with he

Suitability for Computational Genomics

But better methods for using *n*-gra