Generalized edit distance

Reina Käärik

Edit distance

- There are lots of applications where you have to measure the similarity between two strings:
  - Search engines
  - Handwriting recognition
  - Speller checkers
  - ...

- Example: *režiim* vs. *rezhiim*
  and
  režiim vs. *raamat*

Edit distance

- Also known as Levenshtein distance
- The minimal number of edit steps that has to be made to transform one string to another
- Allowed transformations are:
  - Deletion
  - Insertion
  - Replacement
  - Transposition

Edit distance

- Example:
  - *režiim* vs. *rezhiim* – edit distance 2
  - *režiim* vs. *raamat* – edit distance 5

- Also ...
  - *režiim* vs. *riim* – edit distance 2

Generalized edit distance

- The edit distance algorithm that allows to define additional transformations

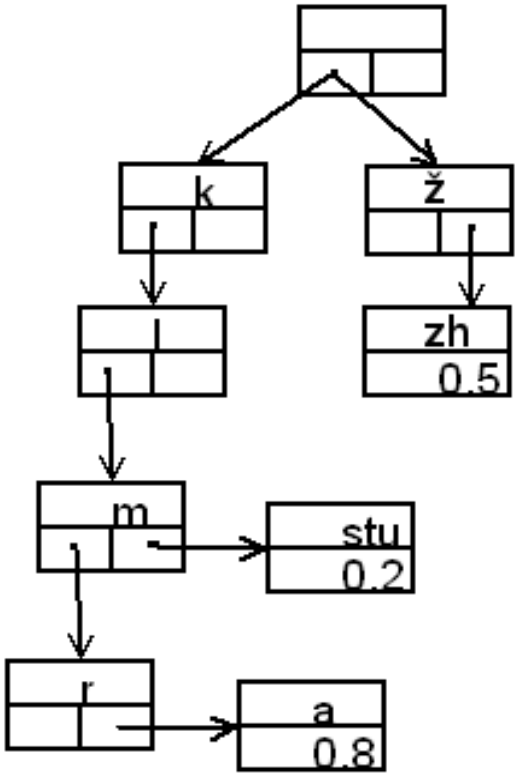
- Example:
  Let's define additional transformation: ž → zh with weight 0.5
  - *režiim* vs. *rezhiim* – edit distance 0.5
  - *režiim* vs. *riim* – edit distance 2

How to make the algorithm faster

- Transformations in trie datastructure
  - Before:

    klm:stu:0.2
    klmr:a:0.8
    ž:zh:0.5

  - After



How to make the algorithm faster

- String set in trie datastructure
- Example:

  tartu vs. *alliksoo*

		a	l	l	i	k	s	o	o
	0	1	2	3	4	5	6	7	8
t	1	1	2	3	4	5	6	7	8
a	2	1	2	3	4	5	5	7	8
r	3	2	2	3	4	5	6	7	8
t	4	3	3	3	4	5	6	7	8
u	5	4	4	4	4	5	6	7	8

and

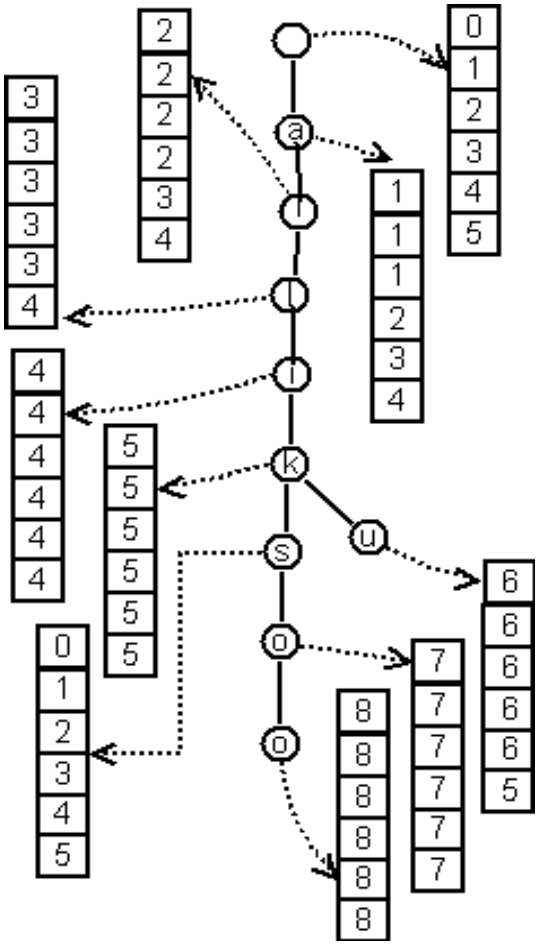
tartu vs. *alliku*

		a	l	l	i	k	u
	0	1	2	3	4	5	6
t	1	1	2	3	4	5	6
a	2	1	2	3	4	5	6
r	3	2	2	3	4	5	6
t	4	3	3	3	4	5	6
u	5	4	4	4	4	5	5

String set in trie datastructure

- Example: *tartu* vs. *alliksoo*

		a	l	l	i	k	s	o	o
	0	1	2	3	4	5	6	7	8
t	1	1	2	3	4	5	6	7	8
a	2	1	2	3	4	5	5	7	8
r	3	2	2	3	4	5	6	7	8
t	4	3	3	3	4	5	6	7	8
u	5	4	4	4	4	5	6	7	8



Finding weights for transformations

- How many times the transformation was used
- How many times the transformation could have been used
- Find the probability and weight

Finding transformations from a set of examples

Questions?