Center, Ctr, Centrum, Zentrum - the challenge of institutional addresses

Holger Schwechheimer[1] and Christine Rimmert[2]

[1] *holger@iwt.uni-bielefeld.de*
University of Bielefeld (Germany)

[2] *christine.rimmert@uni-bielefeld.de*
University of Bielefeld (Germany)

THE TASK: assignment of publications to research facilities and their units

THE PROBLEM: the *addresss* are not standardized, incomplete, incorrect...

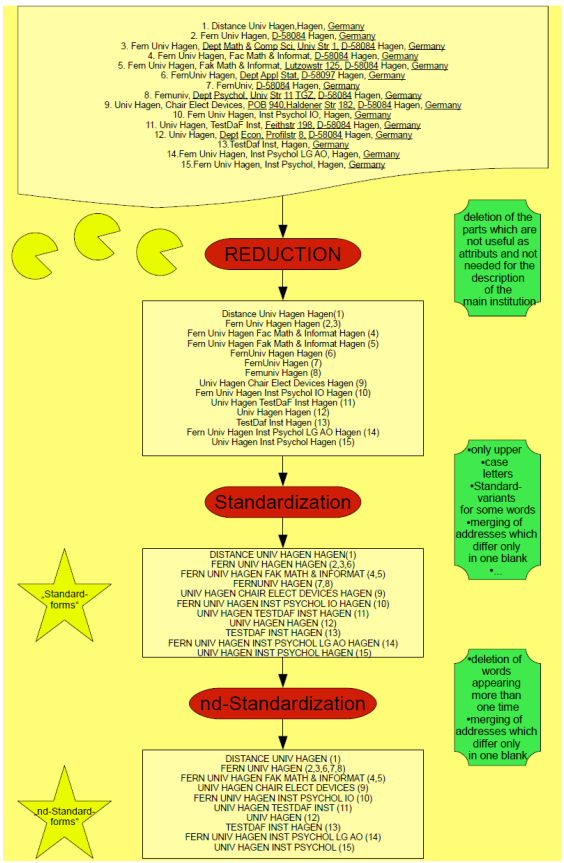
THE POSTER SHOWS: first steps of a semi-automatic approach to standardize the *addresses*

[We started with an example data set taken from WOS: year 2008, country 'Germany' (134.304 address-variants)]

The algorithm consists of the follwing steps:

1. reduction
2. standardization
3. extracting cities
4. extracting tree-structures
5. ...further steps in progress...

Let's accompany 15 examples of addresses on their way through the procedure....



First they get reducedand *standardized* as described above. The *nd-standardization* provides standardized addresses which every single word existsonly once.

To prepare the trees the standardforms and nd-standardforms are stored in table which also contains the word count of the nd-standardforms.

Creating the table (see table below):

For every standardform s* we check if any other standardform *contains* it. If this is the case we write the standardform in the adequate column of the table (depending on the number of words of s*).

Example s*='UNIV HAGEN': two words which appear in all nd-standardforms except 'TESTDAF INST HAGEN'. So we denote 'UNIV HAGEN' in the 2-word-column except for 'TESTDAF INST HAGEN'.

We show only the part of the table up to 4 words because the rest is not very interesting in this example.

standardform	nd-standardform	word count	included 2-words-standardforms	included 3-words-standardforms	included 4-words-standardforms
DISTANCE UNIV HAGEN HAGEN (1)	DISTANCE UNIV HAGEN (1)	3	UNIV HAGEN	DISTANCE UNIV HAGEN	_
FERN UNIV HAGEN HAGEN (2,3,6)	FERN UNIV HAGEN (2,3,6,7,8)	3	UNIV HAGEN	FERN UNIV HAGEN	_
FERN UNIV HAGEN FAK MATH & INFORMAT (4,5)	FERN UNIV HAGEN FAK MATH & INFORMAT (4,5)	7	UNIV HAGEN	FERN UNIV HAGEN	_
FERNUNIV HAGEN (7,8)	FERN UNIV HAGEN (2,3,6,7,8)	3	UNIV HAGEN	FERN UNIV HAGEN	_
UNIV HAGEN CHAIR ELECT DEVICES HAGEN (9)	UNIV HAGEN CHAIR ELECT DEVICES (9)	5	UNIV HAGEN	_	_
FERN UNIV HAGEN INST PSYCHOL IO HAGEN (10)	UNIV HAGEN INST PSYCHOL IO (10)	5	UNIV HAGEN	_	UNIV HAGEN INST PSYCHOL
UNIV HAGEN TESTDAF INST HAGEN (11)	UNIV HAGEN TESTDAF INST (11)	4	UNIV HAGEN	TESTDAF INST HAGEN	UNIV HAGEN TESTDAF INST
UNIV HAGEN HAGEN (12)	UNIV HAGEN (12)	2	UNIV HAGEN	_	_
TESTDAF INST HAGEN (13)	TESTDAF INST HAGEN (13)	3	_	TESTDAF INST HAGEN	_
FERN UNIV HAGEN INST PSYCHOL LG AO HAGEN (14)	FERN UNIV HAGEN INST PSYCHOL LG AO (14)	7	UNIV HAGEN	FERN UNIV HAGEN	UNIV HAGEN INST PSYCHOL
UNIV HAGEN INST PSYCHOL HAGEN (15)	UNIV HAGEN INST PSYCHOL (15)	5	UNIV HAGEN	FERN UNIV HAGEN	UNIV HAGEN INST PSYCHOL

Building trees:

All (different) entries of the table for which the following condition is true will provide a root of a tree: an entry e* (containing n words) of the (4.-6.column of the) table is a root if there is a row in which e* is the entry of the n-word-column and all p-word-columns are empty for p<n.

[Notation-remark: To simplify notation we say 'node st' (where st is a nd-standardform) instead of 'the node belonging to the standardform st' from now on.

In our example we have two roots(and therefore we get two trees):

- 'UNIV HAGEN' and
- 'TESTDAF INST HAGEN'

Building the tree with root 'UNIV HAGEN':

Nodes: We collect all k-word-entries (k=3,...,7) of rows which contain 'UNIV HAGEN' in their 2-word-column. They will be nodes in the tree.

Connections:

- connect all **3-word-nodes** with the root.
- For every **k-word-node (k>3)** check: Does it contain one of the (k-1)-nodes?

Yes*:* connect it with this (k-1)-node and stop the procedure

[If there is more than one (k-1)-word-node contained in the node we take copies of the node and connect in each case a copy with a contained (k-1)-word-node (this makes sure that the received graph has no circles)]

No*:* go to the (k-2)-nodes and proceed analogous and so on up to the root

If no connections are found connect the node with the root.

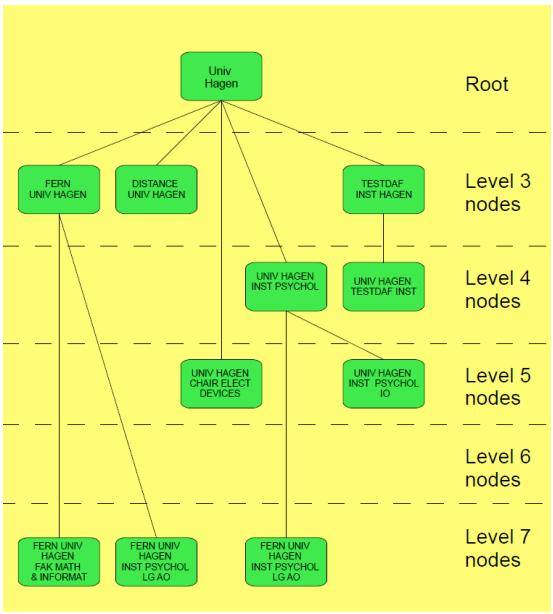
The tree with root

'TESTDAF INST HAGEN'

consists only of the root itself.

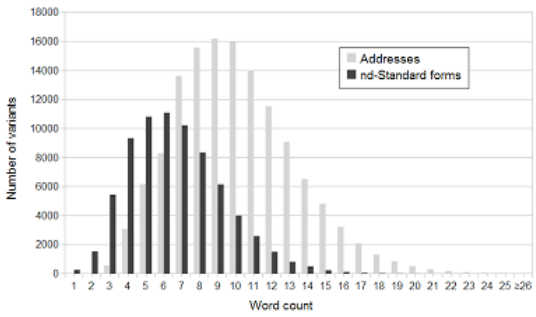
'TESTDAF INST HAGEN' appears as inner node in a tree and also as root in another tree. This is a hint for the following situation:

'TESTDAF INST HAGEN' is a main institution (because it is *complete* and it is a root) and the address(es) which gave us the inner node in the tree with root 'UNIV HAGEN' contain(s) two main institutions (UNIV HAGEN and TESTDAF INST HAGEN). A co-operation between UNIV HAGEN and TESTDAF INST HAGEN may be given.



Now lets us have a look at the complete data source (WOS, Germany, 2008). Standardization and reduction lead to the transformation of the data

concerning number of variants and word count shown in the diagram.



Building *k-trees* up to k=5 we already receive a large *coverage* of standardforms:

k	N (trees)	N (nd)	Percent of total (nd), cumulated
2	1.470	47.054	64
3	3.357	10.805	79
4	3.213	6.989	89
5	2.156	3.311	93