Data consolidation and cleaning using fuzzy string comparisons with -*matchit*- command

2016 Swiss Stata Users Group meeting

Julio D. Raffo
Senior Economic Officer
WIPO, Economics & Statistics Division

Bern
November 17, 2016

Outline

1. What kind of problems *-matchit-* can solve?
2. How to use *-matchit-*? A practical guide
3. Improving performance (speed & quality)
4. Other uses for *-matchit-*

What kind of problems -*matchit*- can solve?

1. When one dataset has duplicated entries which are not uniform

When there is no unique id for observations, inconsistencies arise from:

- Name misspellings — "Thomas Edison" vs. "Tomas Edison"
- Name permutation — "Edison, Thomas " vs. "Thomas Edison"
- Name alternative spellings — "Thomas A. Edison" vs. "Thomas Alva Edison"
- Homonyms — "Thomas Edison Sr." vs. "Thomas Edison Jr."
- Company structure and geography — "Canadian GE" vs. "General Electric"
- Company legal status — "GE inc." vs. "GE co."

2. When merging two different datasets that have no compatible keys

- Same cases than #1, but multiplied by 2
- In practice #1 is a particular case of #2

3. Other uses (we'll discuss these briefly at the end)

- Text similarity scores to be used as variables
- Bags of words

WIPO
WORLD
INTELLECTUAL PROPERTY
ORGANIZATION

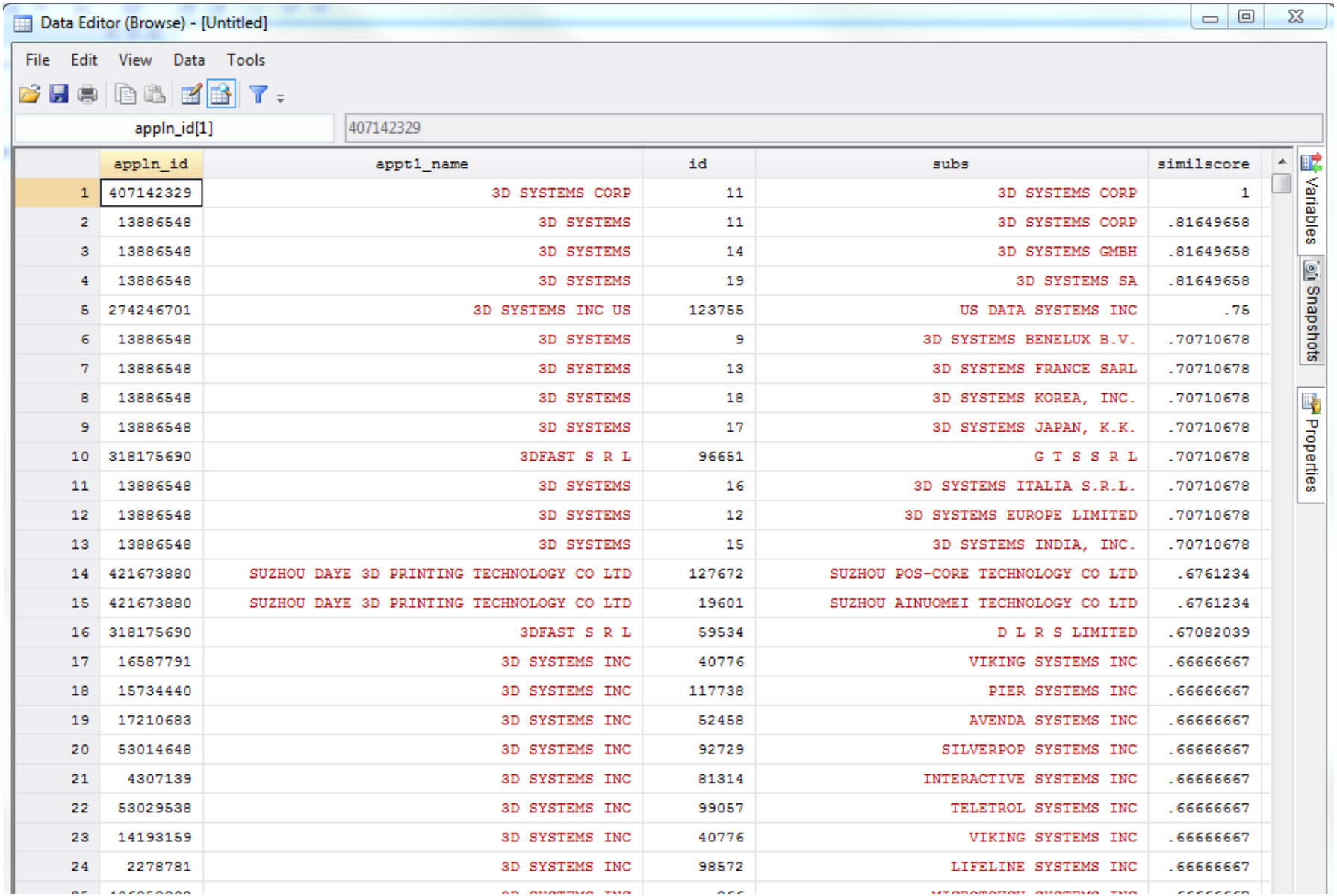
Methods

- Vectoral decomposition of texts
  - Default: Bigram = Splits text into grams of 2 moving chars
    e.g. "John Smith“ splits to Jo oh hn n_ _S Sm mi it th
  - 15+ other built-in methods, including phonetic and hybrids
    e.g. soundex or tokenwrap
- Weighting of vector’s elements
  - Default : no weights (*i.e. all grams =1*)
  - 3 built-in based on *grams* frequency
- Similarity scoring
  - Default: Jaccard = $\langle s_1, s_2 \rangle / |s_1||s_2|$
  - Other 2 built-in functions

A practical guide to use *-matchit-* (1)

```
ssc install matchit // only if not installed already

use file1.dta
matchit id1 txt1 using file2.dta, idu(id2) txtu(txt2)
br

// if you want to manually check results
gsort -similscore

// if you want to use other variables to disambiguate results
joinby id1 using file1
joinby id2 using file2

// Delete what you don't want to match
drop if similscore<.7
drop if addr1!=addr2

save bridge1to2.dta
```

Output: a bridge dataset


Data Editor (Browse) - [Untitled]

File Edit View Data Tools

appln_id[1] 407142329

	appln_id	appt1_name	id	subs	similscore
1	407142329	3D SYSTEMS CORP	11	3D SYSTEMS CORP	1
2	13886548	3D SYSTEMS	11	3D SYSTEMS CORP	.81649658
3	13886548	3D SYSTEMS	14	3D SYSTEMS GMBH	.81649658
4	13886548	3D SYSTEMS	19	3D SYSTEMS SA	.81649658
5	274246701	3D SYSTEMS INC US	123755	US DATA SYSTEMS INC	.75
6	13886548	3D SYSTEMS	9	3D SYSTEMS BENELUX B.V.	.70710678
7	13886548	3D SYSTEMS	13	3D SYSTEMS FRANCE SARL	.70710678
8	13886548	3D SYSTEMS	18	3D SYSTEMS KOREA, INC.	.70710678
9	13886548	3D SYSTEMS	17	3D SYSTEMS JAPAN, K.K.	.70710678
10	318175690	3DFAST S R L	96651	G T S S R L	.70710678
11	13886548	3D SYSTEMS	16	3D SYSTEMS ITALIA S.R.L.	.70710678
12	13886548	3D SYSTEMS	12	3D SYSTEMS EUROPE LIMITED	.70710678
13	13886548	3D SYSTEMS	15	3D SYSTEMS INDIA, INC.	.70710678
14	421673880	SUZHOU DAYE 3D PRINTING TECHNOLOGY CO LTD	127672	SUZHOU POS-CORE TECHNOLOGY CO LTD	.6761234
15	421673880	SUZHOU DAYE 3D PRINTING TECHNOLOGY CO LTD	19601	SUZHOU AINUOMEI TECHNOLOGY CO LTD	.6761234
16	318175690	3DFAST S R L	59534	D L R S LIMITED	.67082039
17	16587791	3D SYSTEMS INC	40776	VIKING SYSTEMS INC	.66666667
18	15734440	3D SYSTEMS INC	117738	PIER SYSTEMS INC	.66666667
19	17210683	3D SYSTEMS INC	52458	AVENDA SYSTEMS INC	.66666667
20	53014648	3D SYSTEMS INC	92729	SILVERPOP SYSTEMS INC	.66666667
21	4307139	3D SYSTEMS INC	81314	INTERACTIVE SYSTEMS INC	.66666667
22	53029538	3D SYSTEMS INC	99057	TELETROL SYSTEMS INC	.66666667
23	14193159	3D SYSTEMS INC	40776	VIKING SYSTEMS INC	.66666667
24	2278781	3D SYSTEMS INC	98572	LIFELINE SYSTEMS INC	.66666667

Variables Snapshots Properties

A practical guide to use *-matchit-* (2)

```
ssc install matchit // only if not installed already

use file1.dta
matchit id1 txt1 using file2.dta, idu(id2) txtu(txt2)
br

// if you want to manually check results
gsort -similscore

// if you want to use other variables to disambiguate results
joinby id1 using file1
joinby id2 using file2

// Delete what you don't want to match
drop if similscore<.7
drop if addr1!=addr2

save bridge1to2.dta
```

A practical guide to use *-matchit-* (3)
(one dataset)

```
ssc install matchit // only if not installed already

use file1.dta
matchit id1 txt1 using file1.dta, idu(id1) txtu(txt1)

// Delete what you don't want to match

// in case of one dataset only
keep id*
gen long new_id = _n
reshape long id, i(new_id) j(n)
ssc install group_id // if not installed (by Robert Picard)
group_id new_id , matchby(id)
```

Output for one dataset: potential pairs

Data Editor (Browse) - [Untitled]

File Edit View Data Tools

var7[592]

	appln_id	appt1_name	appln_id1	appt1_name1	similscore
10	13886548	3D SYSTEMS	15814681	3D SYSTEMS INC	.81649658
13	407142329	3D SYSTEMS CORP	23860364	3D SYSTEMS INC	.66666667
126	274246701	3D SYSTEMS INC US	54170929	3D SYSTEMS INC	.8660254
231	14909511	3D SYSTEMS INC	55509565	3D SYSTEMS INC A CALIFORNIA CO	.70710678
243	49420281	3D SYSTEMS INC	274246701	3D SYSTEMS INC US	.8660254
269	55509565	3D SYSTEMS INC A CALIFORNIA CO	8468816	3D SYSTEMS INC	.70710678
464	274246701	3D SYSTEMS INC US	13040959	3D SYSTEMS INC	.8660254
506	13886548	3D SYSTEMS	14193577	3D SYSTEMS INC	.81649658
545	14305765	3D SYSTEMS VALENCIA	16120045	3D SYSTEMS INC	.66666667
546	37689	3D SYSTEMS INC	14305765	3D SYSTEMS VALENCIA	.66666667
589	4307154	3D SYSTEMS INC	55509565	3D SYSTEMS INC A CALIFORNIA CO	.70710678
592	48992787	3D SYSTEMS INC	274246701	3D SYSTEMS INC US	.8660254
597	47848618	3D SYSTEMS INC	53333827	3D SYSTEM INC	.66666667
621	274246701	3D SYSTEMS INC US	14239398	3D SYSTEMS INC	.8660254
733	13886548	3D SYSTEMS	315829601	3D SYSTEMS INC	.81649658
743	415613	3D SYSTEMS INC	13886548	3D SYSTEMS	.81649658
830	48726825	3D SYSTEMS INC	407142329	3D SYSTEMS CORP	.66666667
845	274246701	3D SYSTEMS INC US	47131653	3D SYSTEMS INC	.8660254

A practical guide to use *-matchit-* (4)
(one dataset)

```
ssc install matchit // only if not installed already

use file1.dta
matchit id1 txt1 using file1.dta, idu(id1) txtu(txt1)

// Delete what you don't want to match

// in case of one dataset only
keep id*
gen long new_id = _n
reshape long id, i(new_id) j(n)
ssc install group_id // if not installed (by Robert Picard)
group_id new_id , matchby(id)
```

How to improve performance?

- Similarity score accuracy:
  - Use built-in ***weights*** to give higher scores to less frequent text
  - Use different built-in ***similmethod***
    token *is better with “cleaner “ data, but worse with misspelled*
  - Use different built-in ***score*** functions
    minsimple *highlights matched,* ***simple*** *highlights unmatched text*

- Computation speed:
  - Remove redundant information
    use ***stopwordsauto*** *and* ***diagnose*** *options*
  - Reduce the size of Index:
    1-gram<2-gram<3-gram<4-gram<soundex<metaphone<token
  - Reduce the depth of Index:
    1-gram>2-gram>3-gram>4-gram>soundex>metaphone>token

```
. matchit appln_id appt1_name using corp.dta, idu(id) txtu(subs) di sim(token)
Matching current dataset with corp.dta
Similarity function: token

Performing preliminary diagnosis
--------------------------------
```

Analyzing Master file
List of most frequent grams in Master file:

	grams	freq	grams_per_obs
1.	LTD	4179	0.1421
2.	INC	3354	0.1140
3.	CO	3057	0.1039
4.	CORP	2676	0.0910
5.	GMBH	1746	0.0594
6.	AG	1574	0.0535
7.	TECHNOLOGY	1250	0.0425
8.	UNIV	1208	0.0411
9.	&	1019	0.0346
10.	SYSTEMS	1013	0.0344
11.	IND	994	0.0338
12.	ELECTRIC	938	0.0319
13.	3D	738	0.0251
14.	STEEL	708	0.0241
15.	INST	668	0.0227
16.	HEAVY	581	0.0198
17.	KK	574	0.0195
18.	MOTOR	556	0.0189
19.	TECHNOLOGIES	546	0.0186
20.	OPTICAL	523	0.0178

Analyzing Using file
List of most frequent grams in Using file:

	grams	freq	grams_per_obs
1.	LIMITED	22994	0.1673
2.	LTD	13220	0.0962
3.	LLC	10705	0.0779
4.	INC	9876	0.0719
5.	LTD.	7681	0.0559
6.	CO.,	5242	0.0381
7.	CO	4926	0.0358
8.	INC.	4787	0.0348
9.	&	4370	0.0318
10.	SERVICES	4239	0.0308
11.	DE	3991	0.0290
12.	HOLDINGS	3708	0.0270
13.	COMPANY	3520	0.0256
14.	GMBH	3500	0.0255
15.	INTERNATIONAL	3290	0.0239
16.	CO.,LTD.	3022	0.0220
17.	PTY	2867	0.0209
18.	BHD	2610	0.0190
19.	SYSTEMS	2595	0.0189
20.	SDN	2471	0.0180

Usually a good idea to remove these

PERTY

```
. matchit appln_id appt1_name using corp.dta, idu(id) txtu(subs) di sim(token)
Matching current dataset with corp.dta
Similarity function: token

Performing preliminary diagnosis
--------------------------------
Analyzing Master file
List of most frequent grams in Master file:
                grams    freq    grams_per_obs
  1.              LTD    4179           0.1421
  2.              INC    3354           0.1140
  3.               CO    3057           0.1039
  4.             CORP    2676           0.0910
  5.             GMBH    1746           0.0594
  6.               AG    1574           0.0535
  7.       TECHNOLOGY    1250           0.0425
  8.             UNIV    1208           0.0411
  9.                &    1019           0.0346
 10.          SYSTEMS    1013           0.0344
 11.              IND     994           0.0338
 12.         ELECTRIC     938           0.0319
 13.               3D     738           0.0251
 14.            STEEL     708           0.0241
 15.             INST     668           0.0227
 16.            HEAVY     581           0.0198
 17.               KK     574           0.0195
 18.            MOTOR     556           0.0189
 19.     TECHNOLOGIES     546           0.0186
 20.          OPTICAL     523           0.0178

Analyzing Using file
List of most frequent grams in Using file:
                grams    freq    grams_per_obs
  1.          LIMITED   22994           0.1673
  2.              LTD   13220           0.0962
  3.              LLC   10705           0.0779
  4.              INC    9876           0.0719
  5.             LTD.    7681           0.0559
  6.              CO.,   5242           0.0381
  7.               CO    4926           0.0358
  8.             INC.    4787           0.0348
  9.                &    4370           0.0318
 10.         SERVICES    4239           0.0308
 11.               DE    3991           0.0290
 12.         HOLDINGS    3708           0.0270
 13.          COMPANY    3520           0.0256
 14.             GMBH    3500           0.0255
 15.    INTERNATIONAL    3290           0.0239
 16.         CO.,LTD.    3022           0.0220
 17.              PTY    2867           0.0209
 18.              BHD    2610           0.0190
 19.          SYSTEMS    2595           0.0189
 20.              SDN    2471           0.0180
```

Usually a good idea to remove these

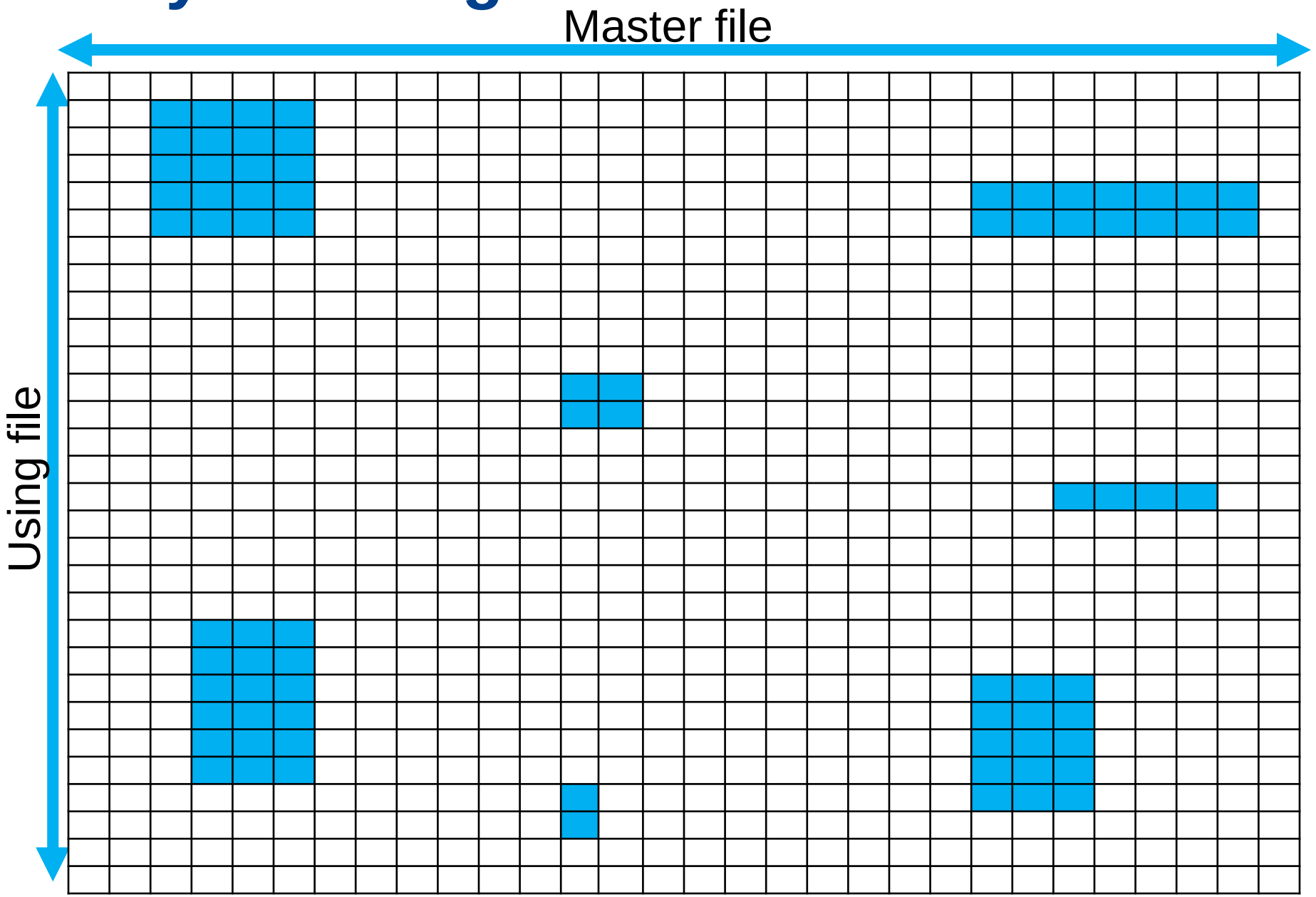
PERTY

Overall diagnosis
Pairs being compared: Master(29415) x Using(137451) = 4.043e+09
Estimated maximum reduction by indexation (%):98.63
(note: this is an indication, final results may differ)

List of grams with greater negative impact to indexation:
(note: values are estimated, final results may differ)

	grams	crosspairs	max_common_space	grams_per_obs
1.	LTD	55246380	1.37	0.1043
2.	INC	33124104	0.82	0.0793
3.	CO	15058782	0.37	0.0478
4.	GMBH	6111000	0.15	0.0314
5.	CORP	5231580	0.13	0.0278
6.	&	4453030	0.11	0.0323
7.	LLC	3778865	0.09	0.0663
8.	SYSTEMS	2628735	0.07	0.0216
9.	TECHNOLOGY	2580000	0.06	0.0199
10.	AG	1163186	0.03	0.0139
11.	TECHNOLOGIES	661206	0.02	0.0105
12.	LIMITED	528862	0.01	0.1379
13.	DE	522821	0.01	0.0247
14.	ELECTRIC	485884	0.01	0.0087
15.	OF	462660	0.01	0.0139
16.	PTY	412848	0.01	0.0180
17.	SA	345871	0.01	0.0141
18.	HITACHI	302162	0.01	0.0070
19.	COMPANY	285120	0.01	0.0216
20.	STEEL	284616	0.01	0.0067

Why indexing?



Checking performance of index

```
. use pat.dta, clear
. matchit appln_id appt1_name using corp.dta, idu(id) txtu(subs)
Matching current dataset with corp.dta
Similarity function: bigram
Loading USING file: corp.dta
Indexing USING file.
0%
20%
40%
60%
80%
Done!
Computing results
        Percent completed ...    (search space saved by index so far)
        20%               ...    (97%)
        40%               ...    (97%)
        60%               ...    (97%)
        80%               ...    (97%)
        Done!
Total search space saved by index: 97%
```

Other uses of -matchit-

- *-matchit-* can also be applied to two variables of the same dataset

```
use file1.dta
matchit id1 txt1 using file2.dta, idu(id2) txtu(txt2)
joinby id1 using file1
joinby id2 using file2
* drop if addr1!=addr2
// let's use column syntax instead
matchit addr1 addr2, g(addrsimil)
drop if addrsimil<.7
```

- This can also be applied as an alternative to indexation
- *-freqindex-* (included with *-matchit-)* can be used to generate "bag of words" or custom weight files
- All functions (similarity, score or weights) can be easily customized by users

WIPO
WORLD
INTELLECTUAL PROPERTY
ORGANIZATION

Thank you!

julio.raffo@wipo.int