Creating LRs with FSTs
Part VII

Syntax, etc.

Mans Hulden
(University of Helsinki)
Iñaki Alegria
(University of The Basque Country)

Applications

Identification of entities
- Dates, numbers, named entities

Surface syntax (after POS tagging)
- Noun phrases, verb phrases...

Translation of dates, numbers...

Transfer of phrases/chains in MT

...

foma

New operators

- Longest matching and insertion
- @-> substitution of longest matched
- ... matched string

OriginalString **@->** TagBegin ... TagEnd ;

Identifying entities

```
define Char [a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z] ;
define Capital [A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z] ;
define NumberElem ["0"|1|2|3|4|5|6|7|8|9] ;
define NumberSymbol ["."|","] ;
define EntiElem Capital [ Capital | Char ]* ;

define EntiStr EntiElem [(" ") EntiElem]* ;
define TagEnti [EntiStr @-> "<ENTI>" ... "</ENTI>"] ;

define NumberStr NumberElem [(NumberSymbol) NumberElem]*;
define TagNumber [NumberStr @-> "<NUMB>" ... "</NUMB>"];

regex TagEnti .o. TagNumber;
```

Identifying entities

Result from foma

```
...
5.1 kB. 9 states, 253 arcs, Cyclic.
April 14, 2010 - Nelson Mandela was honoured
<ENTI>April</ENTI> <NUMB>14</NUMB>, <NUMB>2010</NUMB> -
   <ENTI>Nelson Mandela</ENTI> was honoured
```

English dates

```
...
define Day [(1|2) Number | 3 "0" | 3 1];
define Year Number (Number) (Number) (Number);
define RegDates [WeekDay | Month " " Day (", " Year)];
define DateParser [RegDates @-> "<DATE>" ... "</DATE>"];

defined DateParser: 4.5 kB. 17 states, 238 arcs, Cyclic.
4.5 kB. 17 states, 238 arcs, Cyclic.
April 14, 2010 - Nelson Mandela was honoured
<DATE>April 14, 2010</DATE> - Nelson Mandela was honoured
```

http://www.stanford.edu/~laurik/fsmbook/examples/DateParser.html

Translating numbers to word sequences

[transl_numbers (slightly simplified)]

```
# INPUT: 2,001,634
define Tag1 "," -> M || _ Number^3 .#. ;
define Tag2 "," -> M M || _ Number^3 M ;
define Tag3 Number -> ... C || _ Number^2 [M|.#.] ;
define Tag4 Number -> ... X || _ Number [M|.#.] ;
#2MM0C0X1M6C3X4

define NtoW1 1 X "0"->"ten",1 X 1->"eleven" || _ [M|.#.];
define NtoW2 2->"twenty",3->"thirty"    || _ X ;
define NtoW3 1->"one", 2->"two", 3->"three" || _ [C|M|.#.];

define End1 M M -> " million " ;
define End2 M -> " thousand " ;
define End3 C -> " hundred " ;
define End4 X -> 0 ;
```

Testing the translation rules

```
...
3.6 kB. 27 states, 206 arcs, Cyclic.
2,001,634
2MM0C0X1M6C3X4
34.7 kB. 178 states, 2157 arcs, Cyclic.
two million one thousand six hundred and thirty-four
...
```

Transferring verb chains

Used in a MT system

For modal/periphrastic verbs

FSTs are not for changing word order, but

Identification and substitution of patterns are possible

4 steps

- Identifying and tagging classes of verb chains (based on analyses in source language)
- Including the pattern in the target language corresponding to the class
- Substitution/insertion of the elements in the pattern using information from the source analysis
- Cleaning the result

Transferring verb chains: example

he tenido que:

he/haber/1P/Perf tenido/tener que (ir) (Spanish)

→

(joan) behar izan edun/1P/Perf (Basque)

Eskerrik Asko!
Kiitos!

Foma can help you, and you can help *foma*:

http://foma.sf.net

We can help you with *foma* (and your language):

Mans Hulden: mans.hulden@helsinki.fi
Iñaki Alegria: i.alegria@ehu.es