Converting the "Enciclopèdia Catalana" bilingual MRD to an MTD

L. Benítez
G. Escudero
J. Farreres
G. Rigau

Report LSI–98–3–T

Converting the "Enciclopèdia Catalana" bilingual MRD to an MTD.

L. Benítez, G. Escudero, J. Farreres, G. Rigau

Contents

1 Introduction

Our aim is to retrieve semi-automatically lexical knowledge from bilingual MRDs (Machine Readeable Dictionaries) [Atserias et al. 97, Benítez et al 98a,b].Concretelly we are dealing with an English–Catalan bilingual dictionary (both directions) [DEC 96].

Obviously a MRD does not provide an immediate source of lexical knowledge. Complex processes are necessary in order to convert MRDs to MTDs (Machine Tractable Dictionaries) [Rigau 98]. Usually, this process takes profit from typographical information, that is, using this typographical marks we can make explicit several fields from the MRD. This process is described in this report.

The MRD is coded in SGML in order to make our task easier. As each direction uses different marks to code the same attributes, we have a different treatment for each one. For instance,

English–Catalan

```
<E>a </E><T><v>ar</v> un <v>m</v>, una <v>f</v>.
<i>A man</i>, un home. <i>A woman</i>, una dona;
(<i>rate, price, etc</i>) a, per.
<i>Three times a week</i>, tres vegades per setmana.
<i>Thirty pounds a month</i>, trenta lliures el mes</T>
```

Abbreviations are coded in italics, and grammatical and semantic ones are mixed. To make both directions of the bilingual equivalent in structure, grammatical and semantic abbreviations must be separated. In appendix A.1 are the abbreviations resulting from our treatment.

Catalan–English

```
<E>a </E><T><i>prp</i> (<i>lloc</i>) in.
<i>Viu a Barcelona</i>, she lives in Barcelona;
(<i>direcci</i>) to.
<i>Vaig anar a Anglaterra</i>, I went to England;
(<i>temps</i>) in, at. <i>A la nit</i>, at night;
(<i>complement indirecte</i>) to.
<i>Vaig donar el diari a la mare</i>,
I gave the newspaper to my mother</T>
```

Abbreviations are coded in italics, and the ones referred to gramatical information are coded distinct from the abbreviations referred to semantics. In appendix B.1 are the abbreviations resulting from our treatment.

Marks

The meaning of each mark are the following:

$\langle E\rangle \ldots \langle /E\rangle$ Refers to the entry in both directions.

$\langle T\rangle \ldots \langle /T\rangle$ Refers to the translation of the entry.

$\langle v\rangle \ldots \langle /v\rangle$ Refers to italics in the "English–Catalan" dictionary.

$\langle i\rangle \ldots \langle /i\rangle$ Refers to italics in the "Catalan–English" dictionary.

2 Preparing information

We decided to mark and distinguish all abbreviations that can be detected. Although only the grammatical codes were necessary. We have choosen the following four marks and categories:

CG grammar category

CM morfologic code

CS semantic code

REG use

and we have assigned each abbreviation to one and only one category. See appendix A.1 and B.1 for a complete list of these abbreviations.

2.1 Grammatical information $\langle CG \rangle \ldots \langle CG \rangle$

Catalan–English

```
<E>abadessa </E> <T><i><CG>f</CG></i> abbess</T>
<E>abandonar </E> <T><i><CG>vt</CG></i> to abandon, leave</T>
<E>abandonar </E> <T> <i><CG>vp</CG></i> to abandon osf</T>
<E>abans </E> <T><i><CG>av aj</CG></i> before</T>
<E>abast </E> <T><i><CG>m</CG></i> reach, range</T>
```

English–Catalan

```
<E>abdicate </E> <T><v><CG>v tr</CG></v> abdicar</T>
<E>absent </E> <T><v><CG>adj</CG></v> absent</T>
<E>back </E> <T> <v><CG>v intr</CG></v> recular</T>
<E>envelope </E> <T><v><CG>n</CG></v> sobre</T>
```

2.2 Morfological information $\langle CM \rangle \ldots \langle CM \rangle$

Catalan–English

```
<E>caserna </E> <T><i><CG>f</CG></i> barracks <i><CM>pl</CM></i></T>
<E>duana </E> <T><i><CG>f</CG></i> customs <i><CM>pl</CM></i></T>
<E>golfes </E> <T><i><CG>fpl</CG></i> loft <i><CM>sg</CM></i></T>
<E>noces </E> <T><i><CG>fpl</CG></i> wedding <i><CM>sg</CM></i></T>
```

English–Catalan

```
<E>abbey </E> <T><v><CG>n</CG></v> abadia <v><CM>f</CM></v></T>
<E>abdomen </E> <T><v><CG>n</CG></v> abdomen <v><CM>m</CM></v></T>
<E>hair </E> <T><v><CG>n</CG></v> cabells <v><CM>pl</CM></v></T>
<E>scarlet </E> <T><v><CG>adj</CG></v> escarlata <v><CM>inv</CM></v></T>
```

2.3 Semantic information $\langle CS \rangle \ldots \langle CS \rangle$

Catalan–English

```
<E>acte </E> <T><i><CG>m</CG></i> <i><CS>tea</CS></i> act</T>
<E>be </E> <T><i><CG>m</CG></i> <i><CS>zoo</CS></i> lamb</T>
<E>bus </E> <T><i><CG>m</CG></i> <i><CS>aut</CS></i> bus</T>
<E>cel </E> <T><i><CG>m</CG></i> <i><CS>rlg</CS></i> heaven</T>
<E>dau </E> <T><i><CG>m</CG></i> <i><CS>jcs</CS></i> dice</T>
<E>pal </E> <T><i><CG>m</CG></i> <i><CS>mar</CS></i> mast</T>
```

English–Catalan

```
<E>aside </E> <T> <v><CG>n</CG></v> <v><CS>teat</CS></v> apart</T>
<E>chalk </E> <T><v><CG>n</CG></v> <v><CS>min</CS></v> creta</T>
<E>cite </E> <T><v><CG>v tr</CG></v> <v><CS>dr</CS></v> citar</T>
<E>credit </E> <T><v><CG>v tr</CG></v> <v><CS>com</CS></v> abonar</T>
<E>dope </E> <T><v><CG>v tr</CG></v> <v><CS>esport</CS></v> dopar</T>
<E>draft </E> <T><v><CG>v tr</CG></v> <v><CS>mil</CS></v> quintar</T>
```

2.4 Registers $\langle REG \rangle \ldots \langle REG \rangle$

Catalan–English

```
<E>casta </E> <T><i><CG>f</CG></i> <i><REG>fg</REG></i> class</T>
<E>fatxa </E> <T><i><CG>f</CG></i> <i><REG>fm</REG></i> look</T>
<E>nu </E> <T><i><CG>aj</CG></i> <i><REG>fg</REG></i> bare</T>
<E>merda </E> <T><i><CG>f</CG></i> <i><REG>vlg</REG></i> shit</T>
```

English–Catalan

```
<E>alibi </E> <T><v><CG>n</CG></v> <v><REG>fam</REG></v> excusa</T>
<E>call </E> <T><v><CG>v tr</CG></v> <v><REG>fig</REG></v> evocar</T>
<E>graft </E> <T> <v><CG>v intr</CG></v> <v><REG>vulg</REG></v>
pencar</T>
<E>ice </E> <T><v><CG>n</CG></v> <v><REG>US</REG></v> nevera
<v><CM>f</CM></v></T>
<E>ice </E> <T><v><CG>n</CG></v> <n>ice-box</n> <v><REG>UK</REG></v>
congelador <v><CM>m</CM></v></T>
```

3 Main decisions extracting information

We have two files with the information prepared as described in section 2, one file for the "English–Catalan" dictionary and the other for the "Catalan–English" one. In this section we only describe the process for nouns.

Collocations, definitions, morphological information and gender inflection have a specific process. In this section we explain the method used to extract the pairs formed by a word and its translation, their link to WordNet, and some further treatment.

In both directions of the dictionary:

Collocations:

Some entries correspond to collocations, so the definition does not correspond to a simple word but a combination of words. We have to take that into account to retrieve more precise information.

```
<E>agent </E> <T> <n>agent de canvi i borsa</n> stockbroker</T>
<E>cambra </E> <T> <n>cambra de bany</n> bathroom</T>
<E>dibuix </E> <T> <n>dibuixos animats</n> cartoon</T>
<E>escola </E> <T> <n>escola bressol</n> kindergarten</T>
<E>joc </E> <T> <n>fora de joc</n> offside</T>
<E>lluna </E> <T> <n>lluna de mel</n> honeymoon</T>

<E>age </E> <T> <n>old age</n> vellesa <v>f</v></T>
<E>bad </E> <T> <n>bad habit</n> <v>n</v> vici <v>m</v></T>
<E>canine </E> <T> <n>canine tooth</n> <v>n</v> ullal <v>m</v></T>
<E>day </E> <T> <n>day nursery</n> <v>n</v> guarderia <v>f</v></T>
<E>early </E> <T> <n>early morning</n> <v>n</v> matinada <v>f</v></T>
<E>family </E> <T> <n>family name</n> <v>n</v> cognom <v>m</v></T>
```

Definitions given by other entries:

Some entries refer to another entry in the same language. These ones are separated from the rest to be treated afterwards. These words are given the same definition as the referred words.

```
<E>au </E> <T><i>f</i> = <n>ocell</n></T>
<E>caldo </E> <T><i>m</i> = <n>brou</n></T>
<E>escarxofa </E> <T><i>f</i> = <n>carxofa</n></T>
<E>larinx </E> <T><i>f</i> = <n>laringe</n></T>
<E>medecina </E> <T><i>f</i> <i>fm</i> = <n>medicament</n></T>
<E>naturalesa </E> <T><i>f</i> = <n>natura</n></T>
<E>quadro </E> <T><i>m</i> = <n>quadre</n></T>
<E>rabosa </E> <T><i>f</i> = <n>guineu</n></T>

<E>antiquarian </E> <T><v>n</v> = <n>antiquary</n></T>
<E>exam </E> <T><v>n</v> = <n>examination</n></T>
<E>laundromat </E> <T><v>n</v> = <n>launderette</n></T>
<E>liter </E> <T><v>n</v> = <n>litre</n></T>
<E>maths </E> <T><v>pl</v> = <n>mathematics</n></T>
<E>pretense </E> <T><v>n</v> = <n>pretence</n></T>
<E>sulfur </E> <T><v>n</v> = <n>sulphur</n></T>
<E>tzar </E> <T><v>n</v> = <n>tsar</n></T>
```

Definitions of more then one entry or collocation:

Some definitions can be also considered as definitions from another word or collocation. We can detect that case by (*or*...) or (*o*...) depending on the language source. We save a copy of that entries to give them the same definition as the main word or collocation.

```
<E>barret </E> <T> <n>barret de copa</n> (o <n>barret de mitja copa</n>)
top hat </T>

<E>bath </E> <T> (or <n>bathtub</n>) <v>n</v> banyera <v>f</v></T>
<E>diesel </E> <T>(or <n>diesel oil</n>) <v>n</v> gasoli <v>m</v></T>
<E>elm </E> <T>(or <n>elm tree</n>) <v>n</v> om <v>m</v></T>
```

In the "English–Catalan" dictionary:

Where to place morphological information of catalan words

When we find morphological information of catalan words we code *english_word catalan_word#morph_info*, and when we have a catalan collocation we code *english_word catalan_collocation#morph_info*

```
<E>abbess </E> <T><v>n</v> abadessa <v>f</v></T>

abbess abadessa#f

<E>abbot </E> <T><v>n</v> abat <v>m</v></T>

abbot abat#m

<E>sea </E> <T><v>n</v> mar <v>m/f</v></T>

sea mar#m/f

<E>battle</E><T><n>battlefield</n><v>n</v> camp <v>m</v> de batalla</T>

battlefield camp_de_batalla#m

<E>cockroach </E> <T><v>n</v> escarabat <v>m</v> de cuina</T>

cockroach escarabat_de_cuina#m
```

In the "Catalan–English" dictionary:

***What to do whith gender inflection*:**

Consider these entries:

```
acomodador -a @m usher
acomodador -a @f usherette
actor -triu @m actor
actor -triu @f actress
```

note that outwardly there are same entries appearing twice. In fact they differ because of the gender (@f means femenine, @m means masculine). When entries refer to masculine, we remove the information about femenin gender. So we'll store:

```
acomodador @m usher
actor @m actor
```

On the other way, when entries refer to femenine, we build the word corresponding to the femenin gender. For instance:

```
acomodador -a @f usherette  -->  acomodadora @f usherette
gros -ossa @f thumb         -->  grossa @f thumb
hereu -eva @f heiress       -->  hereva @f heiress
nebot -oda @f niece         -->  neboda @f niece
actor -triu @f actress      -->  actriu @f actress
criat -ada @f servant, maid -->  criada @f servant, maid
```

More information about femenine gender infelxion is shown in appendix B.2
When we find entries like:

```
ambaixador -a @mf ambassador
```

as they do not give us more precise information we store:

```
ambaixador @m ambassador
```

4 Step by step

4.1 Preprocess

1. Translate from DOS format to Unix format

   ```
   dos2unix catangl.txt.0 catangl.txt.1
   ```

2. Transform every multiple definition to a single one.

```
gawk '$0~"^<E>"{if(entry!="")print entry;entry=$0;next}
{entry=entry " " $0}
END{print entry}' catangl.txt.1 > catanl.txt.2
```

3. Identify every single definition.

```
gawk '{gsub(";","</T><T>",$0);print$0}' catangl.txt.2
> catangl.txt.3
```

4. Put one line per definition.

```
gawk '{gsub(" ","_",$0);print$0}' catangl.txt.3 |
gawk '{gsub("<T>"," <T>",$0);print$0}' |
gawk '{for(i=2;i<=NF;i++)print $1,$i}' |
gawk '{gsub("_"," ",$0);print$0}' > catangl.txt.4
```

4.2 English–Catalan

1. Building the lexical source.

```
procesAC1.pl < anglcat.txt.4 > anglcat.font_lexica
```

2. Keeping information to be used in the next process.

   (a) Files with information about entries with '=' i '(or '

   ```
   procesAC1.pl < anglcat.txt.4 |
   procesAC2.pl | egrep '\(or <n>' > angcat.or

   procesAC1.pl < anglcat.txt.4 |
   procesAC2.pl | egrep '=' > angcat.igual
   ```

   (b) We create file 'AC.or' from file 'angcat.or'. File 'AC.or' contains the same links as main entries.

   ```
   procesAC5.pl < angcat.or | sort -u |
   join - word_syns >  AC.or
   ```

   (c) Generation of 'AC.igual' with *old_word new_word*

   ```
   procesAC4.pl < angcat.igual  >  AC.igual
   ```

3. Linking English words and its WN1.5 synsets to Catalan words.

```
procesAC1.pl < anglcat.txt.4 | procesAC2.pl | procesAC3.pl |
sort -u | join - word_syns | egrep -v '=' |
cat - AC.or | sort -u  >  wn_AC.noms
```

```
join AC.igual wn_AC.noms | gawk '{print $2,$3,$4}' |
cat - wn_AC.noms | sort -u > wn-AC.noms

procesCA6.pl < wn-CA.noms > wn_CA.noms.def
```

4.3 Catalan–English

1. Building the lexical source.

```
procesCA1.pl < catangl.txt.4 > catangl.font_lexica
```

2. Keeping information to be used in the next proces.

   (a) Files with information about entries with '=' i '(or '.

```
procesCA1.pl < catangl.txt.4 |
procesCA2.pl | egrep '\(o ' > catang.or

procesCA1.pl < catangl.txt.4 |
procesCA2.pl | egrep '=' > catang.igual
```

   (b) We create file 'CA.or' from file 'catang.or'. File 'CA.or' contains the same links as main entries.

```
procesCA5.pl < catang.or | sort -u |
gawk '{print $2,$1}'  | join - word_syns >  CA.or
```

   (c) Generation of 'CA.igual' with *old_word new_word.*

```
procesCA4.pl < catang.igual  > CA.igual
```

3. Linking Catalan words to English words and its WN1.5 synsets.

```
procesCA1.pl < catangl.txt.4 | procesCA2.pl | procesCA3.pl |
gawk '{print $2,$1}' | egrep -v '=' | egrep -v '\.' |
egrep -v '^ ' | sort -u | join - word_syns |  cat - CA.or |
sort -u  >  wn_CA.noms

gawk '{gsub("\@"," \@",$0);print$2,$3,$1,$4}' wn_CA.noms |
sort -u | join - CA.igual | gawk '{print$3,$5,$2,$4}' |
gawk '{gsub(" \@","\@",$0);print$0}' | cat - wn_CA.noms |
sort -u > wn-CA.noms

procesCA6.pl < wn-CA.noms > wn_CA.noms.def
```

4.4 Final processing

Generation of the homogeneous file from both sources.

```
cat wn_AC.noms.def wn_CA.noms.def | sort -u > wn_dict.noms
```

A English–Catalan dictionary

A.1 Abbreviations

GRAMMATICAL CATEGORY

adj	adj adv	adj n	adj pron	adv	adv adj prep
adv conj	adv n	adv prep	ar	conj	conj adv
interj	n	n pron	prep	pron	v intr
v tr	v tr intr				

SEMANTIC FILE

aeron	agr	agr med	anat	arquit	art
art fotog	art lit	astr	aut	biol	biol tecn
bot	bot elect quím	cin	cin etc	cin fotog	cin teat
com	dr	econ	elect	elect mil	esport
ferroc	fotog	fs	gastr	geog	geog elect
gram	hist	inform	lit	mar	mar ferroc
mat	med	mil	mil min	mil polt etc	min
ms	ms lit	ms teat	polt	qum	relig
relig mil	teat	teat cin	tecn	txt	zool
zool tecn					

MORPHOLOGICAL CODE

f	f pl	f sing	inv	m	m f
m/f	pl	pp	pt	pt pp	sing
tb pl					

REGISTER

UK	US	atr	desp	fam	fig
fig fam	tb fig	vulg			

A.2 Source code in Perl

See [Wall et al. 96].

ProcesAC1.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {

# Transformacions previes

s/<v>(adj|pl|n|m|pol.t|tb pl|vulg) ([^<]*)<\/v>/<v>$1<\/v> <v>$2<\/v>/g;
s/(<v>n)<\/v> <v>(pron<\/v>)/$1 $2/g;
s/<v>(v) ([in]*tr)( [[in]*tr]*|) ([^<]*)<\/v>/<v>$1 $2$3<\/v> <v>$4<\/v> /g;
s/(<v>v tr)<\/v> <v>(intr<\/v>)/$1 $2 /g;
s/(<v>adv) (esport<\/v>)/$1<\/v> <v>$2/g;
s/(<v>adv) (fam<\/v>)/$1<\/v> <v>$2/g;
s/(<v>adj)<\/v> <v>(n<\/v>) /$1 $2/g;
s/(<v>adj)<\/v> <v>(pron<\/v>) /$1 $2/g;
s/(<v>adj)<\/v> <v>(pron) (pl)(<\/v>) /$1 $2<\/v> <v>$3<\/v>/g;
s/(<v>adj)<\/v> <v>(adv<\/v>) /$1 $2/g;
s/(<\/v>) (<\/T>)/$1$2 /g;

# Categories gramaticals

s/<v>(ad[^<]*)<\/v>/<v><CG>$1<\/CG><\/v>/g;
s/<v>(v [^<]*)<\/v>/<v><CG>$1<\/CG><\/v>/g;
s/<v>(n[^<]*)<\/v>/<v><CG>$1<\/CG><\/v>/g;
s/<v>(con[^<]*)<\/v>/<v><CG>$1<\/CG><\/v>/g;
s/<v>(int[^<]*)<\/v>/<v><CG>$1<\/CG><\/v>/g;
s/<v>(pr[^<]*)<\/v>/<v><CG>$1<\/CG><\/v>/g;
s/<v>(ar)<\/v>/<v><CG>$1<\/CG><\/v>/g;

# Camps semantics

s/<v>(a[egnsu][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(ar[^<][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>([beghlqrz][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>([beghlqrz][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(m[^ \/][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(t[^b][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
```

```
s/<v>(ci[^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(co[^n][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(dr)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(f[eo][^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(f.s)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(info[^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;
s/<v>(po[^<]*)<\/v>/<v><CS>$1<\/CS><\/v>/g;

# Codis Morfologics

s/<v>([fm]\/[^<]*)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>(sing)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>(m|f)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>(m f|f pl|f sing)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>(tb pl)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>(inv[^<]*)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>(p[ptl][^<]*)<\/v>/<v><CM>$1<\/CM><\/v>/g;
s/<v>f fam<\/v>/<v><CM>f<\/CM><\/v> <v><REG>fam<\/REG><\/v>/g;

# Registres

s/<v>(U[^<]*)<\/v>/<v><REG>$1<\/REG><\/v>/g;
s/<v>(desp|vulg|fam)<\/v>/<v><REG>$1<\/REG><\/v>/g;
s/<v>(tb fig)<\/v>/<v><REG>$1<\/REG><\/v>/g;
s/<v>(fig[^<]*)<\/v>/<v><REG>$1<\/REG><\/v>/g;
s/<v>(at[^<]*)<\/v>/<v><REG>$1<\/REG><\/v>/g;


 if (/<CG>([^<]*)<\/CG>/) {
  s/<CG>([^<]*)<\/CG>/$catgram=$1/e;
  s/<v>$catgram<\/v>/<v><CG>$catgram<\/CG><\/v>/g;
 }
 else {
  s/<T>/<T><v><CG>$catgram<\/CG><\/v>/g;
 }

 print;
 }
```

ProcesAC2.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {
if(!(/\$/) & (/CG>n/)){
s/\[[^\]]*\]//g;
if(/<n>/) {
s/(<v>[^\(]*<\/v>)[ ]*(<n>[^<]*<\/n>)/$2 $1 /g;
s/<E>[^<]*<\/E>[^<]*<T>[ ]*<n>([^<]*)<\/n>/<E>$1 <\/E> <T>/g;
}
print;
}
# while
}
```

ProcesAC3.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {

s/(<E>([^<]*)[ ]*<\/E>)/$2/g;
s/\(<i>[^<]*<\/i>\)//g;
s/\([^\)]*\)//g;
s/(<CM>f pl<\/CM>)/<CM>f+pl<\/CM>/g;
s/(<CM>m f<\/CM>)/<CM>m+f<\/CM>/g;
s/(<CM>f sing<\/CM>)/<CM>f+sing<\/CM>/g;
s/(<CM>pt pp<\/CM>)/<CM>pt+pp<\/CM>/g;
s/(<CM>tb pl<\/CM>)/<CM>tb+pl<\/CM>/g;
s/( <n>[^<]*<\/n>)//g;
s/(<n>[^<]*<\/n>)//g;
s/( <v><CM>([^<]*)<\/CM><\/v>)/#$2/g;
s/(<v><CM>([^<]*)<\/CM><\/v>)/#$2/g;
s/<v><CG>(\w+) (\w+)<\/CG><\/v>/<v><CG>$1_$2<\/CG><\/v>/g;
s/<v><CG>(\w+) (\w+) (\w+)<\/CG><\/v>/<v><CG>$1_$2_$3<\/CG><\/v>/g;
s/(<v><CG>([^<]*)<\/CG><\/v>)/\@$2/g;
s/(<v><CS>[^<]*<\/CS><\/v>)//g;
s/(<v><REG>[^<]*<\/REG><\/v>)//g;
```

```
s/(<i>[^<]*<\/i>\, [^\.]*\.)//g;
s/(<i>[^<]*<\/i>\, [^<]*<\/T>)/<\/T>/g;
s/\[[^\]]*\]//g;
s/ \@/\@/g;
s/ <T> / <T>/g;
s/ <T> //g;
s/ <T>//g;
s/  <T>//g;
s/<\/T>//g;
s/\@n_pron/\@n/g;
s/\@n\#pl_/\@n /g;
s/\@n\#m_/\@n /g;

if (!(/\$/)) {
if (/\@n/) {

($a,$b,$c,$d) = split('\, ',$_);
 ($primer,$segon) = split('\@n ',$a);
  if (!$segon) { ($primer, $segon) = split('\@n',$a);}
  $primer =~ tr/ /_/;
  $primer =~ s/(.*)_$/$1/;
  $segon =~ s/[ ]*/ /;
  $segon =~ tr/ /_/;
  $segon =~ s/^_(.*)/$1/;
  $segon =~ s/(.*)_$/$1/;
  $segon =~ s/(.*)\.$/$1/;
  $segon =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
  $segon =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
  $segon =~ s/__//g;
  $segon =~ s/_\#//g;
  $aa = join(' ',$primer,$segon);
  $aa =~ s/\@n//g;
  $aa =~ s/ \#pl_/ /g;
  $aa =~ s/\#pl_/\#pl/g;
  $aa =~ s/ \#pl/ /g;
  $aa =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
  $aa =~ s/\#m\#f/#m/;

   print $aa,"\n";
```

```
if ($b) {
 $b =~ s/[ ]+/ /;
 $b =~ tr/ /_/;
 $b =~ s/^_(.*)/$1/;
 $b =~ s/(.*)_$/$1/;
 $b =~ s/(.*)\.$/$1/;
 $b =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
 $b =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
 $b =~ s/([^_]*)(_-[^_]*)/$1/g;
 $b =~ s/__//g;
 $b =~ s/_\#//g;
 $bb = join(' ',$primer,$b);
 $bb =~ s/ \#pl_/ /g;
 $bb =~ s/\#pl_/\#pl/g;
 $bb =~ s/ \#pl/ /g;
 $bb =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
 $bb =~ s/\#m\#f/#m/;
print $bb,"\n";
if ($c) {
 $c =~ s/[ ]+/ /;
 $c =~ tr/ /_/;
 $c =~ s/^_(.*)/$1/;
 $c =~ s/(.*)_$/$1/;
 $c =~ s/(.*)\.$/$1/;
 $c =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
 $c =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
 $c =~ s/([^_]*)(_-[^_]*)/$1/g;
 $c =~ s/__//g;
 $c =~ s/_\#//g;
 $cc = join(' ',$primer,$c);
 $cc =~ s/ \#pl_/ /g;
 $cc =~ s/\#pl_/\#pl/g;
 $cc =~ s/ \#pl/ /g;
 $cc =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
 $cc =~ s/\#m\#f/#m/;
  print $cc,"\n";
```

```
  if ($d) {
   $d =~ s/[ ]+/ /;
   $d =~ tr/ /_/;
   $d =~ s/^_(.*)/$1/;
   $d =~ s/(.*)_$/$1/;
   $d =~ s/(.*)\.$/$1/;
   $d =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
   $d =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
   $d =~ s/([^_]*)(_-[^_]*)/$1/g;
   $d =~ s/__//g;
   $d =~ s/_\#//g;
   $dd = join(' ',$primer,$d);
   $dd =~ s/ \#pl_/ /g;
   $dd =~ s/\#pl_/\#pl/g;
   $dd =~ s/ \#pl/ /g;
   $dd =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
   $dd =~ s/\#m\#f/#m/;
    print $dd,"\n";
}}}

# if linia
}}
# while
}
```

ProcesAC4.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
{
s/<E>([^<]*)<\/E>/$novaentrada=$1/e;
s/<n>([^<]*)<\/n>/$vellaentrada=$1/e;
$novaentrada=~tr/ /_/;
$novaentrada=~ s/(.*)_$/$1/;
$vellaentrada=~tr/ /_/;
$vellaentrada=~ s/(.*)_$/$1/;
print $vellaentrada," ",$novaentrada,"\n";
# while
}
```

ProcesAC5.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {

s/<E>[^<]*<\/E>([^\$]*)\(or <n>([^<]*)<\/n>/<E>$2<\/E> $1\(/g;

s/(<E>([^<]*)[ ]*<\/E>)/$2/g;
s/\(<i>[^<]*<\/i>\)//g;
s/\([^\)]*\)//g;
s/(<CM>f pl<\/CM>)/<CM>f+pl<\/CM>/g;
s/(<CM>m f<\/CM>)/<CM>m+f<\/CM>/g;
s/(<CM>f sing<\/CM>)/<CM>f+sing<\/CM>/g;
s/(<CM>pt pp<\/CM>)/<CM>pt+pp<\/CM>/g;
s/(<CM>tb pl<\/CM>)/<CM>tb+pl<\/CM>/g;
s/( <n>[^<]*<\/n>)//g;
s/(<n>[^<]*<\/n>)//g;
s/( <v><CM>([^<]*)<\/CM><\/v>)/#$2/g;
s/(<v><CM>([^<]*)<\/CM><\/v>)/#$2/g;
s/<v><CG>(\w+) (\w+)<\/CG><\/v>/<v><CG>$1_$2<\/CG><\/v>/g;
s/<v><CG>(\w+) (\w+) (\w+)<\/CG><\/v>/<v><CG>$1_$2_$3<\/CG><\/v>/g;
s/(<v><CG>([^<]*)<\/CG><\/v>)/\@$2/g;
s/(<v><CS>[^<]*<\/CS><\/v>)//g;
s/(<v><REG>[^<]*<\/REG><\/v>)//g;
s/(<i>[^<]*<\/i>\, [^\.]*\.)//g;
s/(<i>[^<]*<\/i>\, [^<]*<\/T>)/<\/T>/g;
s/\[[^\]]*\]//g;
s/ \@/\@/g;
s/ <T> / <T>/g;
s/ <T> //g;
s/ <T>//g;
s/  <T>//g;
s/<\/T>//g;
s/\@n_pron/\@n/g;
s/\@n\#pl_/\@n /g;
s/\@n\#m_/\@n /g;
```

```
if (!(/\$/)) {
if (/\@n/) {

($a,$b,$c,$d) = split('\, ',$_);
 ($primer,$segon) = split('\@n ',$a);
  if (!$segon) { ($primer, $segon) = split('\@n',$a);}
  $primer =~ tr/ /_/;
  $primer =~ s/(.*)(_)+$/$1/;
  $segon =~ s/[ ]*/ /;
  $segon =~ tr/ /_/;
  $segon =~ s/^_(.*)/$1/;
  $segon =~ s/(.*)_$/$1/;
  $segon =~ s/(.*)\.$/$1/;
  $segon =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
  $segon =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
  $segon =~ s/__//g;
  $segon =~ s/_\#//g;
  $aa = join(' ',$primer,$segon);
  $aa =~ s/\@n//g;
  $aa =~ s/ \#pl_/ /g;
  $aa =~ s/\#pl_/\#pl/g;
  $aa =~ s/ \#pl/ /g;
  $aa =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
  $aa =~ s/\#m\#f/#m/;

   print $aa,"\n";

  if ($b) {
   $b =~ s/[ ]+/ /;
   $b =~ tr/ /_/;
   $b =~ s/^_(.*)/$1/;
   $b =~ s/(.*)_$/$1/;
   $b =~ s/(.*)\.$/$1/;
   $b =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
   $b =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
   $b =~ s/([^_]*)(_-[^_]*)/$1/g;
   $b =~ s/__//g;
   $b =~ s/_\#//g;
   $bb = join(' ',$primer,$b);
   $bb =~ s/ \#pl_/ /g;
   $bb =~ s/\#pl_/\#pl/g;
   $bb =~ s/ \#pl/ /g;
   $bb =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
   $bb =~ s/\#m\#f/#m/;
```

```
  print $bb,"\n";
  if ($c) {
   $c =~ s/[ ]+/ /;
   $c =~ tr/ /_/;
   $c =~ s/^_(.*)/$1/;
   $c =~ s/(.*)_$/$1/;
   $c =~ s/(.*)\.$/$1/;
   $c =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
   $c =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
   $c =~ s/([^_]*)(_-[^_]*)/$1/g;
   $c =~ s/__//g;
   $c =~ s/_\#//g;
   $cc = join(' ',$primer,$c);
   $cc =~ s/ \#pl_/ /g;
   $cc =~ s/\#pl_/\#pl/g;
   $cc =~ s/ \#pl/ /g;
   $cc =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
   $cc =~ s/\#m\#f/#m/;
    print $cc,"\n";

  if ($d) {
   $d =~ s/[ ]+/ /;
   $d =~ tr/ /_/;
   $d =~ s/^_(.*)/$1/;
   $d =~ s/(.*)_$/$1/;
   $d =~ s/(.*)\.$/$1/;
   $d =~ s/([^\#]+)(\#[^_]+)_+([^\n]*)/$1_$3$2/g;
   $d =~ s/([^_]*)(_-[^\#]*)\#([^\n]*)/$1\#$3/g;
   $d =~ s/([^_]*)(_-[^_]*)/$1/g;
   $d =~ s/__//g;
   $d =~ s/_\#//g;
   $dd = join(' ',$primer,$d);
   $dd =~ s/ \#pl_/ /g;
   $dd =~ s/\#pl_/\#pl/g;
   $dd =~ s/ \#pl/ /g;
   $dd =~ s/^([^ \#]*)(\#[^_]*)_*/$1 /;
   $dd =~ s/\#m\#f/#m/;
    print $dd,"\n";
}}}

# if linia
}}
# while
}
```

ProcesAC6.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {
s/\#m\+iv/\#m\#inv/g;
s/\#f\+pl/\#fpl/g;
s/\#m\/f/\#mf/g;
s/\#m\#pl/\#mpl/g;
s/\#m\#m/\#m/g;
s/\#m\./\#m/g;
s/\#m\#sing/\#m/g;
s/\#f\+sing/\#m/g;
s/\#f_//g;
s/\#m_//g;
s/\#f\/m/\#mf/g;
print;
# while
}
```

B Catalan–English dictionary

B.1 Abbreviations

GRAMMATICAL CATEGORY

aj	aj av	aj f	aj iv	aj m	aj m av
aj m iv	aj mf	aj pl	aj pr	aj pr ar	aj pr pl
ar	ar f	ar fpl	ar m	ar mpl	av
av aj	av cnj	av m	av prp	av prp m	cnj
f	f/m	fpl	fsg/pl	inj	m
m iv	m/f	mf	mpl	msg/pl	pr
prp	vi	vip	vp	vt	vti
vtip	vtp				

SEMANTIC FILE

aer	agr	agr med	ana	ana bot	ana tcn	arq
arq mar	art	ast	ast tcn	aut	aut frr	bio
bot	cin	cin fot	cin tea	com	dr	ecn
ele	esp	fot	fot cin	frr	fs	fs tcn
geo	grf	grm	grm mat	gst	hst	ifm
jcs	lit	mar	mar aer	mat	mat ana	mat med
med	med txt	mil	min	ms	ms tea	pol
qm	rlg	tb fs	tb geo ecn	tb mat	tb med	tb ms
tb tecn	tcn	tea	tea ms txt	txt grf	zoo	

MORPHOLOGICAL CODE

pl	sg	sg/pl

REGISTER

atr	dsp	fg	fg fm
fg vlg	fm	vlg	

B.2 Femenine gender inflexion

acomodador -a @f usherette
al·lot -a @f girl
benefactor -a @f benefactress
calb -a @f bald patch
característic -a @f characteristic
clos -a @f fence
criptògam -a @f Cryptogamia
dret -a @f right
electrònic -a @f electronics
fer -a @f wild animal, beast
fillol -a @f goddaughter
foll -a @f madwoman
fosc -a @f dark, darkness
fruiter -a @f fruit bowl, fruit dish
gegant -a @f giantess
germanastre -a @f stepsister
graner -a @f broom
impressor -a @f printer
infermer -a @f nurse
jardiner -a @f window box
junt -a @f meeting, assembly
llaurador -a @f ploughwoman
lògic -a @f logic
mecànic -a @f mechanics
menjador -a @f trough, manger
mort -a @f death
mosso -a @f girl
màxim -a @f maxim
músic -a @f music
noi -a @f girl
nét -a @f granddaughter
pastor -a @f shepherdess
pistoler -a @f holster
poltre -a @f filly, foal
polític -a @f policy
porc -a @f sow
pràctic -a @f #pl training #sg
pràctic -a @f pilot
aerodinàmic -a @f aerodynamics
beat -a @f lay sister
besnét -a @f great- granddaughter
cambrer -a @f waitress
caador -a @f hunting jacket
comediant -a @f comedienne
dinàmic -a @f dynamics
dret -a @f to stand up
escultor -a @f sculptress
fill -a @f daughter
flac -a @f weakness
fonètic -a @f phonetics
fresc -a @f fresh air
gat -a @f she-cat
genet -a @f horsewoman
gimnàstic -a @f gymnastics
hidràulic -a @f hydraulics
indirecte -a @f insinuation, hint, allusion
jardiner -a @f gardening
junt -a @f board, council, committee
laic -a @f laywoman
lleter -a @f churn, milk can
matemàtic -a @fsg/pl mathematics, maths
mecànic -a @f mechanism
monjo -a @f nun
mort -a @f death, end
màxim -a @f at most
mímic -a @f mime, mimicry
nen -a @f girl
nàutic -a @f art of navigation
parell -a @f couple
pescador -a @f fisherwoman
plàstic -a @f plastic art, modelling
polèmic -a @f polemic, controversy
polític -a @f politics
prostitut -a @f prostitute
pràctic -a @f in practice
pràctic -a @f practice

quint -a @f class, call-up
raier -a @f raftwoman
segador -a @f harvester, reaper, mower
senyor -a @f lady, wife
travesser -a @f road which crosses a town
tècnic -a @f technique
vell -a @f old woman
viudo -a @f widow
xicot -a @f fiancée
xofer -a @f chauffeuse, driver
òptic -a @f optician's
òptic -a @f point of view
americà -ana @f jacket
ancià -ana @f elederly woman
artesà -ana @f craftswoman
comú -una @f commune
comú -una @f toilet
cunyat -ada @f sister-in-law
desè -ena @f ten
escocès -esa @f Scotswoman
espòs -osa @f wife
exclusiu -iva @f sole right
francès -esa @f Frenchwoman
gasós -osa @f lemonade
gros -ossa @f first prize
hereu -eva @f heiress
holandès -esa @f Dutchwoman
hoste -essa @f hostess
instantani -ània @f snap
manyac -aga @f caress
minyó -ona @f girl
mitjà -ana @f mean
nebulós -osa @f nebula
normatiu -iva @f rules #pl, regulations #pl
ofensiu -iva @f offensive
padrí -ina @f grandmother
pagès -esa @f to turn a deaf ear
paisà -ana @f to be in plain clothes
promès -esa @f fiancée
propietari -ària @f owner, proprietress, landlady
quint -a @f fifth
recte -a @f straight line
sembrador -a @f sower
sogre -a @f mother-in- law
traïdor -a @f traitress, betrayer
vedell -a @f veal
vell -a @f to grow old
xaval -a @f girl, lass
xicot -a @f girl, lass
ètic -a @f ethics
òptic -a @f optics
actor -triu @f actress
amfitrió -ona @f hostess
anglès -esa @f Englishwoman
besavi -àvia @f great-grandmother
comú -una @f in common
criat -ada @f servant, maid
degà -ana @f doyenne
empresari -ària @f businesswoman
espadatxí -ina @f swordswoman
exclusiu -iva @f exclusive
fadrí -ina @f young woman
gallès -esa @f Welshwoman
germà -ana @f sister
gros -ossa @f thumb
heroi -oïna @f heroine
hoste -essa @f fair hostess, stewardess
incisiu -iva @f incisor
irlandès -esa @f Irishwoman
marquès -esa @f marchioness
minyó -ona @f maid
nebot -oda @f niece
negatiu -iva @f denial, refusal
noucasat -da @f recently married woman
padrí -ina @f godmother
pagès -esa @f countrywoman
paisà -ana @f countrywoman
primari -ària @f thinness
promès -esa @f promise
religiós -osa @f nun

rodó -ona @f circle
sacerdot -essa @f priestess
serè -ena @f night dew
tallat -ada @f cut, cutting
veterinari -ària @f veterinary science
rodó -ona @f semibreve
serè -ena @f in the open
soldà -ana @f sultana
trencadís -issa @f breakage
viscós -osa @f viscose

B.3 Source code in Perl

See [Wall et al. 96].

ProcesCA1.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {
# s/<i>(.nj)<\/i> / print $1, "\n"/ge;

# Categories gramaticals

s/<i>([^<])<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>([avm][^<])<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>(a[jrv] \w+|a[jrv] \w+ \w+)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>(m iv)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>(vt\w+)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>([^s][^g].\/\w+)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>([^s]*\/\w+)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>(.pl|.nj)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>(pr)<\/i>/<i><CG>$1<\/CG><\/i>/g;
s/<i>(vip|prp)<\/i>/<i><CG>$1<\/CG><\/i>/g;

# Camps semantics

s/<i>(dr|q.)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(a[^tlv][^e])<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>([zb][^rae].)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>([ghjl][^aol].)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>([rt][^io].)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(tb \w+.\w+)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>([ac][^m][^b] \w+)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(ms [^<]*)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(m[aei].)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(m[aei]. [^f][^i]\w+)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(f[^e]. \w+|g.. \w+|t.. ...)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(pol|ifm)<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(f[or][^c])<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(e[slc][^<])<\/i>/<i><CS>$1<\/CS><\/i>/g;
s/<i>(c[io][^ps])<\/i>/<i><CS>$1<\/CS><\/i>/g;
```

```
# Codis Morfologics

s/<i>(pl|sg)<\/i>/<i><CM>$1<\/CM><\/i>/g;
s/<i>(s\w+\/\w+)<\/i>/<i><CM>$1<\/CM><\/i>/g;

# Registres

s/<i>(fg \w+)<\/i>/<i><REG>$1<\/REG><\/i>/g;
s/<i>(fg|fm|vlg|dsp|atr|US)<\/i>/<i><REG>$1<\/REG><\/i>/g;
s/<i>(tb fg)<\/i>/<i><REG>$1<\/REG><\/i>/g;

if (/<CG>([^<]*)<\/CG>/)  {
s/<CG>([^<]*)<\/CG>/$catgram=$1/e;
s/<i>$catgram<\/i>/<i><CG>$catgram<\/CG><\/i>/g;
}
else {
if (!(/=/)) {
s/<T>/<T><i><CG>$catgram<\/CG><\/i>/g;
}
}

print;
 }
```

ProcesCA2.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {
if(!(/\[[^\]]*\]/) & (/CG>[mf]/)){
if(/<n>/) {
s/(<i>[^\(]*<\/i>)[ ]*(<n>[^<]*<\/n>)/$2 $1 /g;
s/<E>[^<]*<\/E>[^<]*<T>[ ]*<n>([^<]*)<\/n>/<E>$1 <\/E> <T>/g;
}
print;
}
# while
}
```

ProcesCA3.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {

s/(<E>([^<]*) <\/E>)/$entrada=$2/e;
s/(<\/i>)(<i>)/$1 $2/g;
s/\([^\)]*\)//g;
s/\(<i>[^<]*<\/i>\)//g;
s/<i>(<CG>[^<]*<\/CG>)<\/i>/$1/g;
s/(<i>[^<]*<\/i>\, [^\.]*\.)//g;
s/(<i>[^<]*<\/i>\, [^<]*<\/T>)/<\/T>/g;
s/(<i><CS>[^<]*<\/CS><\/i>)//g;
s/(<i><REG>[^<]*<\/REG><\/i>)//g;
s/( <n>[^<]*<\/n>)/ /g;
#s/(<n>[^<]*<\/n>)//g;
s/<CG>(\w+) (\w+)<\/CG>/<CG>$1+$2<\/CG>/g;
s/<CG>(\w+) (\w+) (\w+)<\/CG>/<CG>$1+$2+$3<\/CG>/g;
s/(<CG>([^<]*)<\/CG>)/\@$2/g;
#s/( <i><CM>([^<]*)<\/CM><\/i>)/#$2/g;
s/(<i><CM>([^<]*)<\/CM><\/i>)/#$2/g;
s/\[[^\]]*\]//g;
s/[ ]+<T>[ ]+/ <T>/g;
s/<T>//g;
s/<\/T>//g;
s/[ ]+/ /g;

if(/\@mf/&/ -/){ s/ -[^ ]+ \@mf/ \@m/g;}
if(/\@m/&/ -/){ s/ -[^ ]+ \@m/ \@m/g;}
if(/\@f/&/ -/){
s/([^eo]) -a \@f/$1a \@f/g;
s/([eo]) -a \@f/a \@f/g;
s/ -(ena) \@f/$1 \@f/g;
s/s -(esa) \@f/$1 \@f/g;
s/ -(ana) \@f/$1 \@f/g;
s/ -(ona) \@f/$1 \@f/g;
s/s -(osa) \@f/$1 \@f/g;
s/ -(una) \@f/$1 \@f/g;
s/ -(ina) \@f/$1 \@f/g;
s/eu -(eva) \@f/$1 \@f/g;
s/ac -(aga) \@f/$1 \@f/g;
```

```
s/at -(ada) \@f/$1 \@f/g;
s/ari -(ria) \@f/$1 \@f/g;
s/ani -(nia) \@f/$1 \@f/g;
s/avi -(via) \@f/$1 \@f/g;
s/s -(issa) \@f/$1 \@f/g;
s/os -(ossa) \@f/$1 \@f/g;
s/tiu -(iva) \@f/t$1 \@f/g;
s/siu -(iva) \@f/s$1 \@f/g;
s/s -(osa) \@f/$1 \@f/g;
s/oi -(ona) \@f/$1 \@f/g;
s/te -(essa) \@f/t$1 \@f/g;
s/tor -(triu) \@f/$1 \@f/g;
s/bot -(oda) \@f/b$1 \@f/g;
s/casat -(da) \@f/casa$1 \@f/g;
s/ot -(essa) \@f/ot$1 \@f/g;
}

s/(\@[mf]) \#(pl )/$1\^$2/g;

if(/\@[fm]/){
s/ \#/\#/g;
s/(\@[fm][^ ]*)/$catgram=$1/e;

($a,$b,$c,$d) = split('\, ',$_);

  if (/ \@m /) { ($primer,$segon) = split(' \@m ',$a);
}
  if (/ \@mf /) { ($primer, $segon) = split(' \@mf ',$a);}
  if (/ \@mpl /) { ($primer, $segon) = split(' \@mpl ',$a);}
  if (/ \@m\/f /) { ($primer, $segon) = split(' \@m\/f ',$a);}
  if (/ \@msg\/pl /) { ($primer, $segon) = split(' \@msg\/pl ',$a);}
  if (/ \@m\+iv /) { ($primer, $segon) = split(' \@m\+iv ',$a);}
  if (/ \@f /) { ($primer, $segon) = split(' \@f ',$a);}
  if (/ \@fpl /) { ($primer, $segon) = split(' \@fpl ',$a);}
  if (/ \@fsg\/pl /) { ($primer, $segon) = split(' \@fsg\/pl ',$a);}
  $primer =~ tr/ /_/;
  $segon =~ s/[ ]+/ /;
  $segon =~ tr/ /_/;
  $segon =~ s/^_(.*)/$1/;
  $segon =~ s/(.*)_$/$1/;
  $segon =~ s/(.*)\.$/$1/;
  $segon =~ s/_\#/\#/g;
  $segon =~ s/^to_//g;
```

```
  $primera = join('',$primer,$catgram);
  $aa = join(' ',$primera,$segon);
  print $aa,"\n";

  if ($b) {
   $b =~ s/[ ]+/ /;
   $b =~ tr/ /_/;
   $b =~ s/^_(.*)/$1/;
   $b =~ s/(.*)_$/$1/;
   $b =~ s/(.*)\.$/$1/;
   $b =~ s/_\#/\#/g;
   $b =~ s/^to_//g;
   $bb = join(' ',$primera,$b);
  print $bb,"\n";
  if ($c) {
   $c =~ s/[ ]+/ /;
   $c =~ tr/ /_/;
   $c =~ s/^_(.*)/$1/;
   $c =~ s/(.*)_$/$1/;
   $c =~ s/(.*)\.$/$1/;
   $c =~ s/_\#/\#/g;
   $c =~ s/^to_//g;
   $cc = join(' ',$primera,$c);
    print $cc,"\n";

  if ($d) {
   $d =~ s/[ ]+/ /;
   $d =~ tr/ /_/;
   $d =~ s/^_(.*)/$1/;
   $d =~ s/(.*)_$/$1/;
   $d =~ s/(.*)\.$/$1/;
   $d =~ s/_\#/\#/g;
   $d =~ s/^to_//g;
   $dd = join(' ',$primera,$d);
    print $dd,"\n";
}}}

# if linia
}

# while
}
```

ProcesCA4.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {
s/<E>([^<]*)<\/E>/$novaentrada=$1/e;
s/<n>([^<]*)<\/n>/$vellaentrada=$1/e;
$novaentrada=~tr/ /_/;
$novaentrada=~ s/(.*)_$/$1/;
$vellaentrada=~tr/ /_/;
$vellaentrada=~ s/(.*)_$/$1/;
print $vellaentrada," ",$novaentrada,"\n";
# while
}
```

ProcesCA5.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {

s/<E>[^<]*<\/E>([^\$]*)\(o <n>([^<]*)<\/n>/<E>$2<\/E> $1\(/g;

s/(<E>([^<]*)[ ]*<\/E>)/$2/e;
s/(<\/i>)(<i>)/$1 $2/g;
s/\([^\)]*\)//g;
s/\(<i>[^<]*<\/i>\)//g;
s/<i>(<CG>[^<]*<\/CG>)<\/i>/$1/g;
s/(<i>[^<]*<\/i>\, [^\.]*\.)//g;
s/(<i>[^<]*<\/i>\, [^<]*<\/T>)/<\/T>/g;
s/(<i><CS>[^<]*<\/CS><\/i>)//g;
s/(<i><REG>[^<]*<\/REG><\/i>)//g;
s/( <n>[^<]*<\/n>)/ /g;
#s/(<n>[^<]*<\/n>)//g;
s/<CG>(\w+) (\w+)<\/CG>/<CG>$1+$2<\/CG>/g;
s/<CG>(\w+) (\w+) (\w+)<\/CG>/<CG>$1+$2+$3<\/CG>/g;
s/(<CG>([^<]*)<\/CG>)/\@$2/g;
#s/( <i><CM>([^<]*)<\/CM><\/i>)/#$2/g;
s/(<i><CM>([^<]*)<\/CM><\/i>)/#$2/g;
```

```
s/\[[^\]]*\]//g;
s/[ ]+<T>[ ]+/ <T>/g;
s/<T>//g;
s/<\/T>//g;
s/[ ]+/ /g;

if(/\@mf/&/ -/){ s/ -[^ ]+ \@mf/ \@m/g;}
if(/\@m/&/ -/){ s/ -[^ ]+ \@m/ \@m/g;}
if(/\@f/&/ -/){
s/([^eo]) -a \@f/$1a \@f/g;
s/([eo]) -a \@f/a \@f/g;
s/ -(ena) \@f/$1 \@f/g;
s/s -(esa) \@f/$1 \@f/g;
s/ -(ana) \@f/$1 \@f/g;
s/ -(ona) \@f/$1 \@f/g;
s/s -(osa) \@f/$1 \@f/g;
s/ -(una) \@f/$1 \@f/g;
s/ -(ina) \@f/$1 \@f/g;
s/eu -(eva) \@f/$1 \@f/g;
s/ac -(aga) \@f/$1 \@f/g;
s/at -(ada) \@f/$1 \@f/g;
s/ari -(ria) \@f/$1 \@f/g;
s/ani -(nia) \@f/$1 \@f/g;
s/avi -(via) \@f/$1 \@f/g;
s/s -(issa) \@f/$1 \@f/g;
s/os -(ossa) \@f/$1 \@f/g;
s/tiu -(iva) \@f/t$1 \@f/g;
s/siu -(iva) \@f/s$1 \@f/g;
s/s -(osa) \@f/$1 \@f/g;
s/oi -(ona) \@f/$1 \@f/g;
s/te -(essa) \@f/t$1 \@f/g;
s/tor -(triu) \@f/$1 \@f/g;
s/bot -(oda) \@f/b$1 \@f/g;
s/casat -(da) \@f/casa$1 \@f/g;
s/ot -(essa) \@f/ot$1 \@f/g;
}

s/(\@[mf]) \#(pl )/$1\^$2/g;

if(/\@[fm]/){
s/ \#/\#/g;
s/(\@[fm][^ ]*)/$catgram=$1/e;

($a,$b,$c,$d) = split('\, ',$_);
```

```
  if (/ \@m /) { ($primer,$segon) = split(' \@m ',$a);
}
  if (/ \@mf /) { ($primer, $segon) = split(' \@mf ',$a);}
  if (/ \@mpl /) { ($primer, $segon) = split(' \@mpl ',$a);}
  if (/ \@m\/f /) { ($primer, $segon) = split(' \@m\/f ',$a);}
  if (/ \@msg\/pl /) { ($primer, $segon) = split(' \@msg\/pl ',$a);}
  if (/ \@m\+iv /) { ($primer, $segon) = split(' \@m\+iv ',$a);}
  if (/ \@f /) { ($primer, $segon) = split(' \@f ',$a);}
  if (/ \@fpl /) { ($primer, $segon) = split(' \@fpl ',$a);}
  if (/ \@fsg\/pl /) { ($primer, $segon) = split(' \@fsg\/pl ',$a);}
  $primer =~ tr/ /_/;
  $segon =~ s/[ ]+/ /;
  $segon =~ tr/ /_/;
  $segon =~ s/^_(.*)/$1/;
  $segon =~ s/(.*)_$/$1/;
  $segon =~ s/(.*)\.$/$1/;
  $segon =~ s/_\#/\#/g;
  $segon =~ s/^to_//g;

  $primera = join('',$primer,$catgram);
  $aa = join(' ',$primera,$segon);
  print $aa,"\n";

  if ($b) {
   $b =~ s/[ ]+/ /;
   $b =~ tr/ /_/;
   $b =~ s/^_(.*)/$1/;
   $b =~ s/(.*)_$/$1/;
   $b =~ s/(.*)\.$/$1/;
   $b =~ s/_\#/\#/g;
   $b =~ s/^to_//g;
   $bb = join(' ',$primera,$b);
  print $bb,"\n";
  if ($c) {
   $c =~ s/[ ]+/ /;
   $c =~ tr/ /_/;
   $c =~ s/^_(.*)/$1/;
   $c =~ s/(.*)_$/$1/;
   $c =~ s/(.*)\.$/$1/;
   $c =~ s/_\#/\#/g;
   $c =~ s/^to_//g;
   $cc = join(' ',$primera,$c);
    print $cc,"\n";
```

```
  if ($d) {
   $d =~ s/[ ]+/ /;
   $d =~ tr/ /_/;
   $d =~ s/^_(.*)/$1/;
   $d =~ s/(.*)_$/$1/;
   $d =~ s/(.*)\.$/$1/;
   $d =~ s/_\#/\#/g;
   $d =~ s/^to_//g;
   $dd = join(' ',$primera,$d);
    print $dd,"\n";
}}}

# if linia
}

# while
}
```

ProcesCA6.pl

```
#!/usr/local/bin/perl

while (<STDIN>)
 {
s/\@/\#/g;
s/\#m\+iv/\#m\#inv/g;
s/\#f\^pl/\#f/g;
s/\#m\^pl/\#m/g;
s/\#m\/f/\#mf/g;
s/\#f\/m/\#mf/g;
s/\#fsg\/pl/\#f/g;
s/\#msg\/pl/\#m/g;
s/\#mf\)/\#mf/g;
s/ temps\#m\#inv/ temps\#m/g;
print;
# while
}
```

References

[**Atserias et al. 97**] J. Atserias, S. Climent, J. Farreres, G. Rigau, H. Rodríguez. *Combining Multiple Methods for the Automatic Construction of Multilingual WordNets* – Proceedings of Conference on Recent Advances on – NLP. RANLP 97. Tzigov Chark, Bulgaria, 1997.

[**Benítez et al. 98a**] L. Bentez, G. Escudero, J. Farreres, G. Rigau. *Applying Automatic Methods for the Construction of Multilingual WordNets.*– Technical Report LSI-98-4-T. – LSI Department. Universitat Politcnica de Catalunya, 1998.

[**Benítez et al. 98b**] L. Bentez, S. Cervell, G. Escudero, M. Lpez, G. Rigau, M. Taul (Universitat Politcnica de Catalunya; Universitat de Barcelona) *Methods and tools for building the Catalan WordNet.* – In workshop on Language Resources for European Minority Languages at Conference on Language Resources and Evaluation (LREC).– Granada, Spain, 1998.

[**DEC 96**] *Diccionari bàsic català–anglès anglès–català.* – Enciclopèdia catalana, Barcelona 1996. – ISBN: B.26756-1996.

[**Rigau 98**] G. Rigau. *Automatic Acquisition of Lexical Knowledge from MRDs.* PhD Thesis, Departament de Llenguatges i Sistemes Informàtics.– Universitat Politècnica de Catalunya. – Barcelona, 1998.

[**Wall et al. 96**] WALL, Larry; CHRISTIANSEN, Tom; SCHWARTZ, Randal L. *Programming Perl.* – Second Edition. – Sebastopol: O'Relly & Associates, 1996. – ISBN: 1-56592-149-6.