Second workshop on the Apertium free/open-source machine translation platform

Gema Ramírez-Sánchez
Prompsit Language Engineering, S.L.
www.prompsit.com
Campus UMH. Edifici Quórum III.
Av. de la Universitat, s/n. 03203. Elx (Alacant). Spain

22nd May 2015. Zagreb.

Contents

About this workshop

The materials of this workshop on the "Apertium free/open-source machine translation platform" have been created by Prompsit Language Engineering, S.L., as part of the Abu-MaTran (Automatic Building of Machine Translation) project[1] funded by European Union Seventh Framework Programme FP7/2007-2013 under grant agreement number PIAP-GA-2012-324414. Special thanks to Nikola Ljubešić, Filip Klubička, Barbara Dujmic and Víctor M. Sánchez Cartagena for their help.

[1] www.abumatran.eu

Overview

This guide will be useful to complete the hands-on and hands-up practical exercises you will be working during this workshop. They will follow a quick introduction to each subject aimed at covering the following objectives:

1. Understand how does Apertium work, module by module: we will review Apertium modules to know which information we have before working on rules
2. Understand contrastive analysis for machines: we will review some typical morphological and syntactic contrasts between languages to get in the mood
3. Formalising rules for Apertium: you will understand which operations can be performed inside the structural transfer
4. Understand the transfer module: you will see how rules are defined and applied in different transfer levels
5. Reading, modifying and implementing new rules: you will take a look to some rules, try to modify them and write new ones

1 Apertium

1.1 How does Apertium work?

Apertium is made out of modules that all together produce translations. As we have seen, each module performs an action to the input it receives from the precedent module. But let's see how the input and output of each module looks like. We will pay special attention to the module that comes before the transfer: it ouputs lexical units in source language. This is our working material to start thinking rules.

Task 1. Taking a look to Apertium with apertium-viewer [30 min.]

Apertium-viewer[2] is a tool that shows the translation process in Apertium module by module. To access it:

- Make sure you have Java installed in your computer

[2]Further reading: `http://wiki.apertium.org/wiki/Apertium-viewer`

- Download apertium-viewer.jar and save it to your hard drive: `https://svn.code.sf.net/p/apertium/svn/builds/apertium-viewer/apertium-viewer.jar`
- Double-click on apertium-viewer.jar (or right click on it and open with Java Runtime)
- If this does not work for you, try typing from the command:
  - `wget https://svn.code.sf.net/p/apertium/svn/builds/apertium-viewer/apertium-viewer.jar`
  - `java -jar apertium-viewer.jar`

You will finally see an interface like the one shown below.

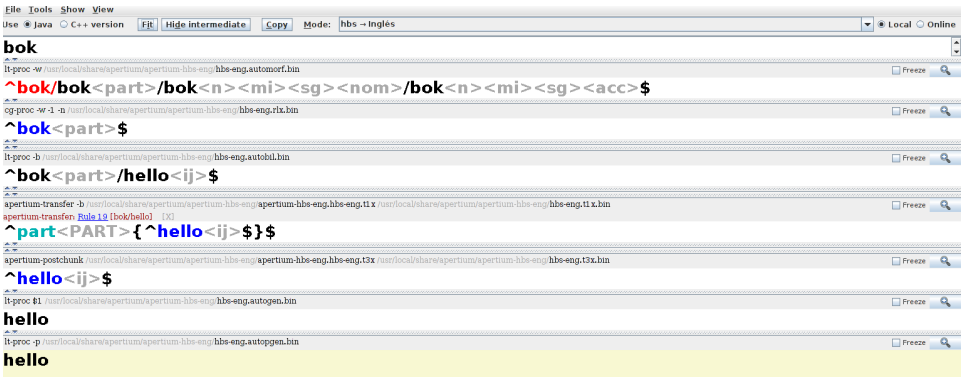

Figure 1: apertium-viewer

Please, follow these instructions:

- First of all, make sure you have the option `Online` (and not `Local`) on the right top of the screen selected. Otherwise click on `Online` and wait for some seconds.
- Next to it you have a menu called `Mode` which says `SELECT A MODE`. In Apertium language a mode is a translation direction. Open the menu and select mode `spanish-galician`. Wait for some seconds, it takes a bit to load all dictionaries...
- For better user experience, let's change the font of the user interface. Go to the menu `View`, click on `Font` and set it to `Dialog-Bold-28`. Then, click on `Done`.

We are ready for testing! Let's start:

- Travelling is always nice, so, let's try with "Have a nice trip" or literally "Good trip" in Spanish: *¡Buen viaje!*
- You will see the translation appearing as you type and the final translation at the end: *Boa viaxe.*
- Even in this simple sentence we have to cope some issues. While you type, having entered just *Buen* and a space, your translation is *Bo*, in masculine and not *Boa* in feminine. Note that *viaje* can be and *noun* or two forms of a *verb*. And in this case it is a noun and femimine, so Apertium has to cope with ambiguitiy and gender here.
- And, how does Apertium know what to do? If you click on any of the bars appearing in the screen and you swipe it down you will start to see all intermediate modules output in Apertium.

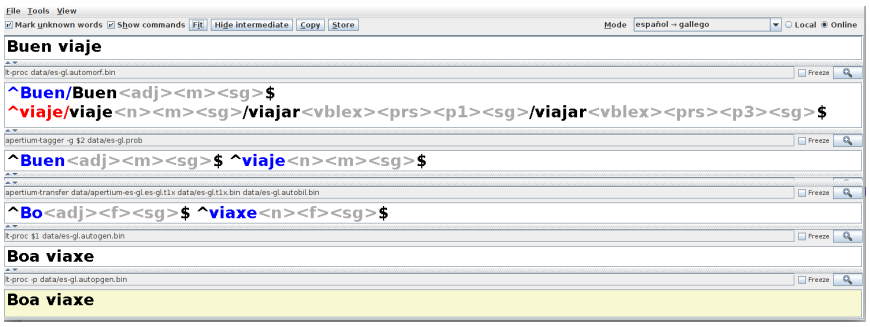

Figure 2: View of apertium-viewer modules

- You will see for each window the command names for each module. You will also find useful to open in a separate tab the wiki page which specifies how part-of-speech and other morphological features[3] are denoted in Apertium:

[3] `http://wiki.apertium.org/wiki/List_of_symbols`

1. **Morphological analyser output**: *lt-proc data/es-gl.automorf.bin*
2. **Part-of-speech tagger**: *apertium-tagger -g $2 data/es-gl.prob*
3. **Multiple-word unit handler**: *apertium-pretransfer*
4. **Transfer**: *apertium-transfer data/apertium-es-gl.es-gl.t1x data/es-gl.t1x.bin data/es-gl.autobil.bin*)
5. **Morphological generator output**: *lt-proc $1 data/es-gl.autogen.bin*
6. **Post-generator output**: *lt-proc -p data/es-gl.autopgen.bin*

- Let's take a look some more sentences.

  Han estado en su casa

  - Inspect module 2 to see ambiguity for *estado* and *casa*.
  - Inspect module 3 to see how ambiguity is solved.
  - Inspect module 5 to see how a compound (synthetic) verb form is turned into a simple (analytic) verb form.
  - Inspect module 5 to see how agreement between *PRONOUN + NOUN* is propagated.
  - Inspect module 6 to see the sequence of lexical forms in the target language: note the mark for the postgenerator (~).
  - Inspect module 8 to see the contraction and the final translation!

  Se llama Xavier, no se llama Mikel

  - Inspect module 2 to see ambiguity and structure of the sentence (in the end)
  - Inspect module 5 to see how the clitic changes position if the noun phrase is positive and not when is negative.
  - Inspect module 9 for final translation.

Before the transfer module, morphological analysis and disambiguation are performed to the source language. These can be problematic and not fully solved. After disambiguation, what we have is a sequence of lexical forms, i.e. lemma, part-of-speech and morphological attributes.

The bilingual phase inside Apertium starts right now!

2 Rules

2.1 Contrastive analysis

One of the key aspects to build a set of transfer rules for Apertium is the result of a contrastive analysis in which the most frequent and systematic phenomena are covered. Let's try to think about it from some bilingual examples in Serbian and Croatian.

Task 2. Contrastive analysis for Serbian and Croatian [30 min.]

Take a look to the following sentences and spot changes that should be **systematically** applied between Serbian [SR] and Croatian [HR]. Put brackets in the source and target sentences to identify these changes. Indicate whether they should be treated in the lexical transfer (LT-) or the structural transfer (ST-). An example is given to you:

- 0-[SR] Njihov izbor ST-[je zamjenica LT-[direktora]] Kosovske policije Atifete Jahjaga.
- 0-[HR] Njihov izbor ST-[zamjenica je LT-[ravnatelja]] Kosovske policije Atifete Jahjaga.

Your turn! Please, record the answers for TASK 2 in the shared spreadsheet under your tab name at `http://tinyurl.com/pwehpvu` [30 min.]

- 1-[SR] Jesen-Petersen je obećao da će se lično angažovati u mobilizaciji podrške donatora za proces povratka.
- 1-[HR] Jessen-Petersen obećao je osobni angažman u mobilizaciji potpore donatora za proces povratka.

- 2-[SR] dodajući da bi poslednja ubistva mogla da utiču na svedoke koji tek treba da svedoče.
- 2-[HR] dodajući kako posljednja ubojstva mogu imati utjecaj na svjedoke koji tek trebaju svjedočiti.

- 3-[SR] On je dodao da Slovačka "trenutno posmatra bezbednost koju je Kosovo ponudilo srpskoj zajednici".
- 3-[HR] Dodao je kako Slovačka "trenutačno promatra sigurnost koju je Kosovo ponudilo srpskoj zajednici".

- 4-[SR] Zemlje jugoistočne Evrope trebalo bi da sarađuju kako bi uspostavile regionalno energetsko tržište[...]
- 4-[HR] Zemlje jugoistočne Evrope trebale bi surađivati kako bi uspostavile regionalno energetsko tržište[...]

- 5-[SR] Protokol imao za cilj stvaranje ≪snažne i ujedinjene opozicije≫,
- 5-[HR] Protokol je za cilj imao uspostavu "snažne i ujedinjene oporbe",

- 6-[SR] U središtu skandala nalazi se kompanija MK Komerc i njen vlasnik Miodrag Kostić.
- 6-[HR] U središtu skandala nalazi se kompanija MK Komerc i njezin vlasnik Miodrag Kostić.

- 7-[SR][...] ubijen je 1. marta 2004. godine.
- 7-[HR] [...]ubijen je 1. ožujka 2004. godine.

- 8-[SR] Očekuje se da će EU u narednih nekoliko meseci doneti odluku o tome da li će Srbiji i Crnoj Gori [...]
- 8-[HR] Očekuje se kako će EU u narednih nekoliko mjeseci odlučiti hoće li Srbiji i Crnoj Gori [...]

- 9-[SR] [...]pozdravljajući sporazum između Beograda i Prištine o bližoj saradnji na polju povratka raseljenih lica.
- 9-[HR] [...]pozdravljajući sporazum Beograda i Prištine o boljoj suradnji u povratku raseljenih osoba.

- 10-[SR] Mada je Molivijatis jasno ukazao da je Kipar njegov prvi prioritet,
- 10-[HR] Iako je Molyviatis jasno istaknuo kako je Cipar njegov prvi prioritet,

- 11-[SR] Ministri inostranih poslova EU će,
- 11-[HR] Ministri vanjskih poslova EU odobrit će,

2.2 Formalising rules

Rules in Apertium are based on a pattern-action structure. Formalising rules consist of defining patterns to be detected, operations to be performed and output for the morphological generator. Let's see how to do it.

Task 3. Formalising Serbian to Croatian rules [30 min.]

Taking the changes that were spotted as structural transfer rules from the previous task, we will now define them as formalised rules. In the same spreadsheet already opened you will find after TASK 3 four columns named:

- **Pattern**: write the pattern of your rule defining all lexical forms involved. For our example above: BITI-clitic + ADJECTIVE + NOUN
- **Checks**: write tests and checks that you consider important for the detected pattern, i.e. agreement checks or conditions to apply one operation or another to it. In our example, checking gender to make sure that there will be agreement between adjective and noun is a must-have.
- **Operations**: write what do you want to do with the pattern in terms of insertions, deletions, substitutions and reorderings. In our example: change the order of the pattern to place the clitic between the adjective and the noun.
- **Output**: write the output(s) of your rule. There might be more than one depending on the checks and operations performed. For our example: ADJECTIVE + BITI-clitic + NOUN

We will discuss the rules together before going further. It you finished, take a look to how the rule in our example would look like in Apertium and try to understand it:

```
<rule comment="Place clitic between adjective and noun">
  <pattern>
  <pattern-item n="CAT__clt_"/>
   <pattern-item n="CAT__adj_"/>
   <pattern-item n="CAT__n_"/>
  </pattern>
  <action>
    <choose>
      <when><!--Check noun gender in source and target-->
        <test>
          <not>
            <equal>
              <clip pos="3" side="sl" part="gender"/>
              <clip pos="3" side="tl" part="gender"/>
            </equal>
          </not>
          <let> <!--If n gender changed, propagate to adj-->
            <clip pos="2" side="tl" part="gender"/>
            <clip pos="3" side="tl" part="gender"/>
          </let>
        </test>
      </when>
      <otherwise>
          <let> <!--If n gender didn't change, keep original gender-->
            <clip pos="2" side="tl" part="gender"/>
            <clip pos="2" side="tl" part="gender"/>
          </let>
      </otherwise>
    </choose>
    <out> <!-- output lexical forms in correct order -->
      <lu>
        <clip pos="2" side="tl" part="whole"/>
      </lu>
      <b pos="1"/> <!-- space -->
      <lu>
        <clip pos="1" side="tl" part="whole"/>
      </lu>
      <b pos="2"/> <!-- space -->
      <lu>
        <clip pos="3" side="tl" part="whole"/>
      </lu>
    </out>
  </action>
</rule>
```

Didn't get the full picture? Don't worry, you will be able to get it after our next step: the structural transfer in Apertium.

2.3 Structural transfer in Apertium

Rules in Apertium were designed to be manually coded by linguists. Most of the language pairs have rules that were manually crafted and implemented. There is another possibility: inferring them automatically from a parallel corpus[4]. We will see now some examples of this technique applied to Serbian and Croatian.

Task 4. Identifying rules and evaluating them [30 min.]

You are presented with a set of sentences in Serbian and the output in Croatian generated by Apertium with transfer rules that have been automatically infefred from a parallel corpus. The patterns and actions applied are already marked in the sentences. You also have a set of possible rules matching them. Identify which rule or rules are applied in each case. Once matched, please answer the following questions: is the rule producing a good output? would you keep it? Would you modify or extend it, how and why? You are asked again to code our answers in the same spreadsheet (look for TASK 4).

- 1-[SR] [...]dobili A-[su specijalno] priznanje žirija.
- 1-[SR2HR] [...]dobili [specijalno su] priznanje žirija.

- 2-[SR] [...], postoje specifični kriterijumi koji bi B-[trebalo] C-[da] budu D-[uzeti u obzir].
- 2-[SR2HR] [...], postoje specifični kriterijumi koji bi [trebao] [kako] budu [uzeti u obzir].

- 3-[SR] [...], E-[američki predsednik] Džordž V. Buš rekao je F-[da] bi Turska G-[trebalo da okonča] upad što je pre moguće.
- 3-[SR2HR] [...], [američki predsjednik] Džordž V. Buš rekao je [kako] bi Turska [trebao okončati] upad što je prije moguće.

- 4-[SR] [...]kako bismo bili u mogućnosti H-[da proizvedemo] više i sami I-[zaradimo] svoje plate[...]

[4]This approach has been explored by researcher Víctor M. Sánchez-Cartagena, Juan A. Pérez-Ortiz and Felipe Sánchez-Martínez, for further information, please read their most recent publication at `http://www.dlsi.ua.es/~fsanchez/pub/pdf/sanchez-cartagena15a.pdf`

- 4-[SR2HR] [...]kako bismo bili u mogućnosti [da proizvedemo] više i sami [zaraditi] svoje plate[...]

- 5-[SR] Troje J-[dodatnih članova] Saveta biće K-[izabrano u parlamentu] L-[posle] otvorenog konkursa[...]
- 5-[SR2HR] Troje [dodatnih članak] Saveta biti će [izabrano u parlamentu] [nakon] otvorenoga konkursa[...]

- 6-[SR] On je M-[dodao da] će to ubrzati evroatlantsku integraciju.
- 6-[SR2HR] On je [kako] će to ubrzati evroatlantsku integraciju.

- 7-[SR] Mnogi N-[smatraju da je] dogovor oko O-[zajedničkog kandidata] [...]
- 7-[SR2HR] Mnogi [smatraju kako je] dogovor oko [zajedničkoga kandidata] [...]

- 8-[SR] Ovaj dijalog bi P-[imao za cilj] da unapredi saradnju,[...]
- 8-[SR2HR] Ovaj dijalog bi [za cilj imao] da unaprijedi suradnju,[...]

- 9-[SR] Pre nego što Q-[počne da obavlja] svoju dužnost,[...]
- 9-[SR2HR] Prije nego koja [počne obavljati] svoju dužnost,[...]

- 10-[SR] , što je izazvalo nacionalnu debatu R-[o tome da li] je možda vreme da se smanji[...]
- 10-[SR2HR] , što je izazvalo nacionalnu debatu [hoće] je možda vrijeme da se smanji[...]

- 11-[SR] Kancelarija glavnog tužioca saopštila je S-[da radi] na izvođenju krivaca pred lice pravde.
- 11-[SR2HR] Kancelarija glavnoga tužitelja saopštila je [raditi] na izvođenju krivaca pred osobu pravde.

- 12-[SR] U intervjuu T-[8. aprila], [...]
- 12-[SR2HR] U intervjuu [8. travnja],[...]

- 13-[SR] PKK daje oprečne cifre U-[i kaže] V-[da] je ubila preko 100 vojnika.
- 13-[SR2HR] PKK daje oprečne cifre [i kazati] [kako] je ubila preko 100 vojnika.

- 14-[SR] Tri dana kasnije Putin je rekao svojoj W-[vladi da pojača] razgovore sa EU [...]
- 14-[SR2HR] Tri dana kasnije Putin je rekao svojoj [vladi pojačati] razgovore s EU [...]

- 15-[SR] Praktično sve kompanije su X-[prisiljene da uračunaju] takve rizike [...]
- 15-[SR2HR] Praktično sve kompanije su [prisiljene uračunati] takve rizike [...]

LETTER	**RULE**	**KEEP**	**COMMENTS**
A	2	yes	extend to biti.clt + adj + noun
B			
C			
D			
E			
F			
G			
H			
I			
J			
K			
L			
M			
N			
O			
P			
Q			
R			
S			
T			
U			
V			
W			
X			

Rule	Pattern	Output
2	*biti.clt + adj*	adj + biti.clt
3	*trebati lp.nt*	trebati lp.m
4.1	*vbmod.pres + da + vblex.pres*	vbmod.pres + vblex.inf
4.2	*vbmod.lp.nt + da + vblex.aor*	vbmod.lp.m + vblex.inf
4.3	*vbmod.lp.pres*	vbmod.lp + vblex.inf
5.1	*vblex.perf.tv.pres*	vblex.perf.tv.inf
5.2	*dati vblex.perf.tv.pres*	kako cnjsub
5.3	*ispuniti vblex.aor*	ispuniti vblex.inf
5.4	*vblex.perf.tv.aor*	vblex.perf.tv.inf
5.5	*dati vblex.perf*	kako cnjsub
6	*dodati + da*	kako cnjsub
8.1	*vblex + pr + cilj n*	pr + cilj n + vblex
8.2	*vblex + do pr + n*	vblex + do + n
8.4	*vblex.perf + pr + n*	vblex.perf + pr + n
9	*vblex.pres + da part + vblex*	vblex.pres + vblex.inf
10	*vblex + adj + vblex*	vblex + adj + vblex
11	*njen prn*	njezin prn
12	*poslije pr*	nakon pr
13	*pr + np*	pr + np.casefrompr
14	*pr + univerzitet n*	pr + sveučilište n
15	*o tome da li*	hoće
16.1	*da + vblex.pres*	vblex.inf
16.2	*da + vblex*	da + vblex
17	*num.f.sg.nom*	num.gen.sg.gen
18.2	*4. num.ord + n.mi.pl.gen*	4. num.ord + n.mi.sg.nom
18.5	*num.ord + n.mi.pl*	num.ord + n.mi.sg
18.6	*num + n*	num + n
19	*np.ant + np.cog*	np.ant.casefromcog + np.cog
20.1	*jul n.pl*	srpanj n.sg
20.2	*član n.ma*	članak n.ma
20.3	*april n.pl*	travanj n.sg
20.4	*june n.pl.gen*	lipanj n.sg.nom
20.5	*univerzitet n.mi.sg.gen*	sveučilište n.nt.pl.acc
20.6	*tok n.ins*	tijekom pr.gen
20.7	*region n.mi*	regija n.f
21.1	*n.m.acc + vblex.perf.iv*	biti biti.clt + n.mi + vblex.perf.iv + održiv
21.2	*n + vblex*	n + vblex
22	*n + izmedu pr + np*	n + np
23	*n + da part + vblex.aor*	n + vblex.inf
24	*n.f + np.ant.f*	n.f.numberandcasefromnp + np.ant.f
25	*n.nt.sg.gen + adj.pst.mi.sg.gen*	n.nt.sg.gen + adj.pst.mi.sg.gen
26.1	*mada cnjsub*	iako cnjsub
26.2	*da cnjsub*	kako cnjsub
28	*cnjcoo + vblex.perf.pres*	cnjcoo + vblex.perf.inf
29	*cnjcoo + vblex.pres + prn*	cnjcoo + vblex.inf + prn
30.1	*inostran adj.ind*	vanjski adj.def
30.3	*godišnji adj.mi*	godišnji adj.nt
31	*adj + da part + vblex.pres*	adj + vblex.inf
32.1	*inostran adj.ind + n*	vanjski adj.def + n
32.2	*adj.pst.ma.sg + n.ma.sg*	adj.pst.ma.sg + n.ma.sg
32.3	*bivši adj + n*	bivši adj + n
32.4	*adj + n*	adj + n
33	*adj + carinski adj*	adj

2.4 Advanced structural transfer

The multiple level transfer in Apertium is normally made of 3 steps: chunker (or transfer), interchunk and postchunk. The best way to understand them is to look to some examples and see how it behaves. The input and output of the whole transfer is again lexical units but in each step this intermediate representation changes to something closer to a shallow syntactic parsing.

Task 5. Inspecting the multiple-level transfer [30 min.].

Open again apertium-viewer and select mode *english-spanish*. This will take some time to download. Please make sure that you have under menu *Show* all three possibilities enabled, specially *Trace transfer and interchunk rules*. You will be prompted to a first example which is not a very interesting one, let's try this one:

Peter's car is broken today.

First of all, you will see some new commands in the process: a little module that was used to handle saxon genitive (now deprecated), and the three-level transfer steps. They look like this :

- **Saxon genitive handler (deprecated)**: *apertium-transfer -n data/apertium-en-es.en-es.genitive.t1x data/en-es.genitive.bin*
- **Transfer**: *apertium-transfer data/apertium-en-es.en-es.t1x data/en-es.t1x.bin data/en-es.autobil.bin*)
- **Second transfer step**: *apertium-interchunk data/apertium-en-es.en-es.t2x data/en-es.t2x.bin*
- **Third transfer step**: *apertium-postchunk data/apertium-en-es.en-es.t3x data/en-es.t3x.bin*

Additionally, you will be able to trace the rules in the chunker (or transfer) and interchunk modules. For our example, if we dismiss the rest of modules and concentrate only in the transfer module as shown in your screen or the picture below.

First we have the **transfer** module: it outputs a noun phrase (*SN*), the genitive mark as a preposition (*pr*), another noun phrase (*SN*), one copulative verb (*Vcop*) and one adverb (*adv*). The rule 117, in which you can

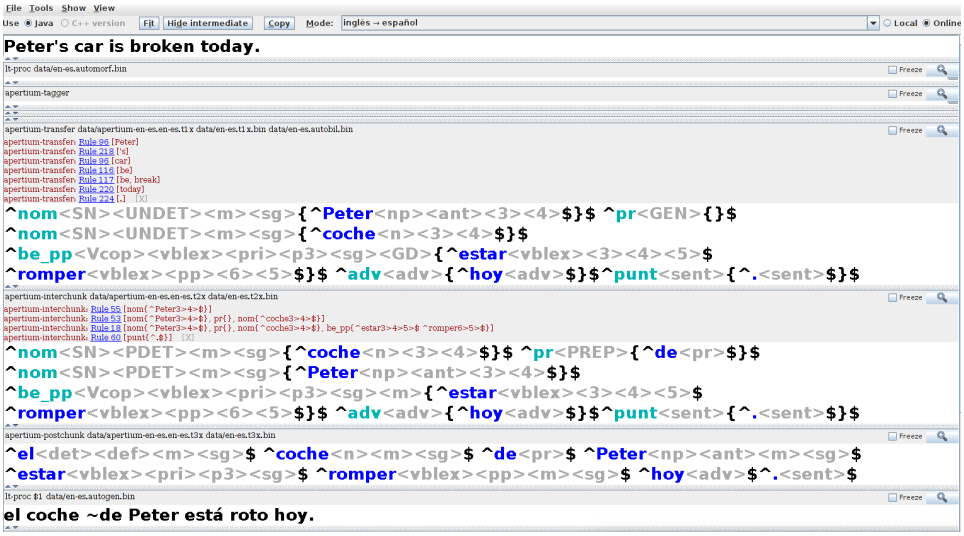

Figure 3: apertium-viewer: English-Spanish

click to inspect it, is the one that generates correctly the translation for the copulative verb. In Spanish it can can be either *ser* or *estar* depending on the meaning of the past participle. Can you read it? Which is the pattern? Which are the conditions and the operations?

Then, we have the **interchunk** module: rule number 18 will take the 2 noun phrases, the preposition and the copulative verb and will make two operations: reordering to generate the correct for of expressing possession in Spanish (*Peter's car* = *Car of Peter*) and agreement between *car* gender and past participle from the copulative verb (*broken* has to be *masculine* as *car* in Spanish). Besides, it tells to the next module that *car* should be determined by an article (*PDET*). If you click on rule 18 you will see it all.

Finally, the **postchunk** will take the information from the previous module and prepend a definite article to *car* but not to *Peter* because is a proper noun. It will output the lexical forms in the target language that after the morphological generator and post-generator will produce the final translation: *El coche de Peter está roto hoy*. Poor guy...

Congrats! By now you are an expert reader of Apertium rules! Writing just will take a bit more effort, maybe for a future workshop!

Recap and useful info

In this workshop you have been introduced to the structural transfer module of the Apertium free/open source machine translation platform.

We thank you for your participation in this workshop and encourage you to join the Apertium community to help us improving. To do so, just subscribe to our mailing list or show up in the chat: we will help you to come in. You will find how to contact us in our wiki page called Contact.[5]

During the Abu-MaTran project, we will go on improving language pairs and interfaces related to Croatian and other languages. We want to hear about you if you have your say about our results. Please contact us through the Abu-MaTran website form.[6]

License

Please contact Gema Ramírez-Sánchez (gramirez at prompsit dot com) for a copy of the source files.

[5] `http://wiki.apertium.org/wiki/Contact`
[6] `http://www.abumatran.eu/?page_id=48`
[7] © Prompsit Language Engineering.