Apertium: A rule-based machine translation platform

Francis M. Tyers

HSL-fakultehta
UiT Norgga árktalaš universitehta
9018 Romsa (Norway)

28th September 2014
Outline

Introduction
  Design
  Development
  Status

Teaching
  Courses
  Google Summer of Code
  Google Code-in

Research
  New language pairs
  Applying unsupervised methods
  Hybrid systems

Future work and challenges
  Challenges
  Plans
  Collaboration
2004 — Spain gets a new government which launches a call for proposals to build machine translation systems for the languages of Spain

The Universitat d’Alacant (UA), in consortium with EHU, UPC, UVigo, Eleka, Elhuyar (Basque Country) and Imaxin (Galicia) get funded to develop two MT systems:

- **Apertium**: Spanish–Catalan, Spanish–Galician
- **Matxin**: Spanish→Basque

Apertium was not built from scratch, but was rather a rewrite of two existing closed-source systems which had been built by UA
Focus

- **Marginalised**: Languages which are on the periphery either societally or technologically (from Breton to Bengali).

- **Lesser-resourced**: Languages for which few free/open-source language resources exist.

- **Closely-related**: Languages which are suited to shallow-transfer machine translation.
Translation philosophy

- Build on word-for-word translation
- Avoid complex linguistic formalisms
  - We like saying “only secondary-school linguistics required”¹
  - Intermediate representation based on morphological information
- Transformer-style systems
  - Analyse the source language (SL), then
  - Apply rules to ‘transform’ the SL to the TL

Step-by-step

Morphological analysis

$ echo "Tiilitalojen keskellä hän kirjoitti minulle nimensä kiinalaisin kirjaimin." | apertium -d . fin-eng-morph
~Tiilitalojen/tiilitalo<n><pl><gen>$ ~keskellä/keskellä<adv>/keskellä<post>$
~hän/hän<prn><pers><sg><nom>$ ~kirjoitti/kirjoittaa<vblex><actv><past><p3><sg>$
~minulle/minä<prn><pers><sg><all>$
~nimensä/nimi<n><pl><nom><pxsp3>/nimi<n><sg><nom><pxsp3>$
~kiinalaisin/kiinalainen<adj><pos><pl><ins>/kiinalainen<adj><v→n><sup><sg><nom>/kiinalainen<n><pl><ins>$
~kirjaimin/kirjain<n><pl><ins>$^./.<sent>$
Morphological disambiguation

$ echo "Tiilitalojen keskellä hän kirjoitti minulle nimensä kiinalaisin kirjaimin." | apertium -d . fin-eng-tagger
^Tiilitalo<n><pl><gen>$ ^keskellä<post>$ ^hän<prn><pers><sg><nom>$
^kirjoittaa<vblex><actv><past><p3><sg>$ ^minä<prn><pers><sg><all>$ ^nimi<n><sg><gen><pxsp3>$
^kiinalainen<adj><pos><pl><ins>$ ^kirjain<n><pl><ins>$^.<sent>$
Step-by-step

```bash
$ echo "Tiilitalojen keskellä hän kirjoitti minulle nimensä kiinalaisin kirjaimin." | apertium -d . fin-eng-biltrans
^Tiilitalo<n><pl><gen>/Brick house<n><pl><gen>$ ^keskellä<post>/in between<pr>$ ^hän<prn><pers><sg><nom>/she<prn><pers><f><sg><nom>/they<prn><pers><mf><pl><nom>/ he<prn><pers><m><sg><nom>$ ^kirjoittaa<vblex><actv><past><p3><sg>/write<vblex><actv><past><p3><sg>/type<vblex><actv><past><p3><sg>/ draft<vblex><actv><past><p3><sg>/spell<vblex><actv><past><p3><sg>/...$
^minä<prn><pers><sg><all>/I<prn><pers><mf><sg><all>$ ^nimi<n><sg><gen><pxsp3>/title<n><sg><gen><pxsp3>$ ^kiinalainen<adj><pos><pl><ins>/Chinese<adj><pos><pl><ins>$ ^kirjain<n><pl><ins>/character<n><pl><ins>$ ^.<sent>$
```
Step-by-step

Lexical selection

$ echo "Tiilitalojen keskellä hän kirjoitti minulle nimensä kiinalaisin kirjaimin." | apertium -d . fin-eng-lextor
~Tiilitalo<n><pl><gen>/Brick house<n><pl><gen>$
~keskellä<post>/in between<pr>$
~hän<prn><pers><sg><nom>/they<prn><pers><p3><mf><sg><nom>$
~kirjoittaa<vblex><actv><past><p3><sg>/write<vblex><actv><past><p3><sg>$
~minä<prn><pers><sg><all>/I<prn><pers><mf><sg><all>$
~nimi<n><sg><gen><pxsp3>/name<n><sg><gen><pxsp3>$
~kiinalainen<adj><pos><pl><ins>/Chinese<adj><pos><pl><ins>$
~kirjain<n><pl><ins>/character<n><pl><ins>$^.<sent>/.<sent>$

Apertium 6 / 41
$ echo "Tiilitalojen keskellä hän kirjoitti minulle nimensä kiinalaisin kirjaimin." | apertium -d . fin-eng-postchunk
^In between<pr>$ ^brick house<n><pl>$ ^they<prn><pers><p3><mf><sg>$ ^write<vblex><past>$ ^to<pr>$ ^I<prn><pers><p1><mf><sg><acc>$ ^name<n><sg>$ ^with<pr>$ ^Chinese<adj>$ ^character<n><pl>$^.<sent>$
Step-by-step

$ echo "Tiilitalojen keskellä hän kirjoitti minulle nimensä kiinalaisin kirjaimin." | apertium -d . fin-eng
In between brick houses they wrote to me name with Chinese characters.

Morphological generation
Modularity

The pipeline architecture makes it straightforward to insert or substitute modules:

- **HFST**:  
  - Replacement for lttoolbox  
  - Used for most “morphologically complex” languages

- **VISL CG**:  
  - Accompanies apertium-tagger  
  - Rule-based morphological disambiguation and shallow-syntactic analysis
What kind of systems are developed?

<table>
<thead>
<tr>
<th>Dissemination</th>
<th>Assimilation</th>
</tr>
</thead>
<tbody>
<tr>
<td>High precision</td>
<td>Low precision</td>
</tr>
<tr>
<td>Min→Min</td>
<td>Gisting</td>
</tr>
<tr>
<td>Maj→Min</td>
<td>Min→Maj</td>
</tr>
<tr>
<td>Closely-related languages</td>
<td>Unrelated or distant languages</td>
</tr>
<tr>
<td>Real <em>translation</em> aids</td>
<td>Reading aids</td>
</tr>
</tbody>
</table>
Trägdårdsföreningen

La Sociedad de Horticultura de Gotemburgo, en sueco Trägdårdsföreningen es un jardín botánico situado en una zona céntrica de la ciudad sueca de Gotemburgo, no lejos de la Estación Central y del parque Brunnsparken.

El jardín propiamente dicho posee una extensión de pocas hectáreas, y está contiguo y adosado al Jardín Botánico de Gotemburgo y sin solución de continuidad a la gran zona verde del parque Slottsskogen, que entonces sumaría una extensión de 175 ha (unos 430 acres), estando incluido el Arboretum y la reserva natural.

Puerta de entrada al Trägdårdsföreningen de Gotemburgo.
Máilmmi vuosttas eamiálbmogiid Pride-festivála ordnejuvvo Gironis

Sámi Nuorra ja Queering Sápmi -prošeakta ordnejit máilmmi vuosttas eamiálbmogiid Pride-festivála golggotmánus Gironis. Festivála sisttisdoallá earret eará ovttastallama, bargobájjid, logaldallamiid ja kulturdoaluid.

Váldooddasat

Sápmi 8.30 0 0
Sovkina i bala – giitá sámiid veahkis

Sii geat ná lähtej mii vuostá ja hehtjej mii sáamamis díkkor koarhkinni, síi dat baket, Guolódaga sámi parltameanitt ságuðoall Valentina Sovkina daja maŋŋebárga New Yorkas.

Sápmi 14:56 0 0
Áile Javo ON:s: Árktalaš guovlu vuordá, ahte loahppadokumeantta boaðusin ruvkelágat rievdaduvvojit

Sámiindí ságuðoall Áile Javo doalai árktalaš guovlu virgjálaš sáhkvaroru máilmmikonferençsas eanaruuqoiatuohtsagastallamis New Yorkas maŋŋebárga.

Sápmi 12:04 0 0
Dutkamuš: Eanandoallit nagodedje ealihiit eambo mániid go sápmelaččat

Šíbit- ja eanandoalltu ealihiid 1700-1900-logun buorebbit go árbevilloh bivu ja boozodoallu. Don dihte suopmelaččat lassánde kjotánwabbot go sápmelaččat, çájohá sámuovlu girkogirjijid vuodul.
Verdenens første urfolks Pride-festivalen ordnes i Kiruna

Samens Ung og Queering Sameland -prosjektet ordner verdenens de første urfolkenes Pride-festivalen i oktober i Kiruna. Festivalen inneholder blant annet samværet, bargobájíid, forelesningene og kulturarrangementet.

Effort

How much effort does it take to develop a system with Apertium?

- 2 weeks minimum
  - Fastest system made (Macedonian→English)
- After 3 months, success rate is around 50%
  - Around half of the students that participate in the Google Summer of Code are able to finish their systems in 12 weeks
- After 6 months, perhaps 75% success rate
  - Students who don’t quite make it in the Google Summer of Code can have their projects picked up and finished in about three months more.
- After 1 year, ...
  - When asking for funding, this is probably the minimum amount of time you’d put.
Success factors

**Technical:**
- How much data is freely available
  - In terms of dictionaries, rule descriptions and corpora
- Stability and compatibility of tagsets (intermediate representation)

**Linguistic:**
- Morphological complexity of the languages
- Genetic and structural similarity of the languages

**Personal:**
- Experience of the developer with the tools
- Staying power of the developer
  - Do they have the staying power to check 100s of lexicon entries/day for weeks on end?
Language pairs
Language pairs

2006

spa -> cat
fra
eng
glg
por

spa

oci

fra

eng

por
Language pairs

2007
Language pairs

2008

spa

cat

fra

oci

eng

glg

por

epo

cym

eus
Language pairs
Language pairs

2012
Language pairs

2013
Language pairs
Major users

- La Voz de Galicia
- La Generalitat de Catalunya
- Ofis Publik ar Brezhoneg
- WikiMedia
- Oslo School District 😊
Online users
Online translation statistics

- 53% - nob-nno
- 20% - spa-cat
- 4% - cat-spa
- 5% - por-spa
- 4% - nno-nob
- 3% - eng-spa
- 3% - spa-eng
- Other - 9%
Outline

Introduction
  Design
  Development
  Status

Teaching
  Courses
  Google Summer of Code
  Google Code-in

Research
  New language pairs
  Applying unsupervised methods
  Hybrid systems

Future work and challenges
  Challenges
  Plans
  Collaboration
Courses organised

- Luxembourg, May 2011
- Shupashkar, January 2012
- Helsinki, May 2013
Luxembourg, May 2011

- 2-day course, funded by the European Commission DGT
- Held at the European Commission
- Organised for translators
- Produced an 80-page course book used in later courses
Luxembourg, May 2011

- 2-day course, funded by the European Commission DGT
- Held at the European Commission
- Organised for translators
- Produced an 80-page course book used in later courses
- Translators not ever so interested in making MT systems 😊
5-day course, funded by Apertium
Held in Shupashkar (Cheboksary), Chuvashia
8 modules, covering all aspects of the system
3 teachers
30 participants from Russia
At least two participants have gone on to work on MT
Helsinki, May 2013

- Course organised at the Dept. Linguistics
- 3 ECTS credits!
- Around 20 students
- Uralicists and linguistics students
Google Summer of Code

What is the Google Summer of Code?

- Offers stipends ($5,500, 12 weeks) for students
- Open-source projects propose 3-month project ideas
- Students apply for projects

Apertium in the Google Summer of Code:

- Around 60 completed projects
- Projects can be:
  - Language pairs
  - Engine development
  - Interfaces
  - Research work
- Massive organisation effort
## Important projects

<table>
<thead>
<tr>
<th>Year</th>
<th>Project Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009</td>
<td>Norwegian Bokmål–Nynorsk</td>
</tr>
<tr>
<td>2010</td>
<td>HFST integration</td>
</tr>
<tr>
<td>2011</td>
<td>–</td>
</tr>
<tr>
<td>2012</td>
<td>Kazakh–Tatar</td>
</tr>
<tr>
<td>2013</td>
<td>Apertium for Android</td>
</tr>
<tr>
<td>2014</td>
<td>Finite-state constraint grammar</td>
</tr>
<tr>
<td></td>
<td>Assimilation evaluation toolkit</td>
</tr>
</tbody>
</table>
Outcomes

- Student retention rate: around 10%
- Over 10 released language pairs
- Approx. €5,000 / year for project funds!
  - Sending students to conferences
  - Organising workshops
  - Funding limited non-research work
Google Code-in

What is the Google Code-in?

- Competition organised for 13–17 year olds
- Organised as ‘tasks’
  - One task should take experienced developer 2 hours
  - Each task is worth one point
- Two kids from each organisation get fully-paid trip to SF

Apertium in the Google Code-in:

- Thousands of completed tasks
- Tasks can be:
  - Language pairs
  - Engine development
  - Interfaces
  - Research work
- Massive organisational effort
Outcomes

- Afrikaans–Dutch language pair
- Support for compound words in lttoolbox
- Free tagged corpus of English WP articles (30,000 words)
- Python interface to Apertium
- New web site
Outcomes

- Afrikaans–Dutch language pair
- Support for compound words in lttoolbox
- Free tagged corpus of English WP articles (30,000 words)
- Python interface to Apertium
- New web site
Outcomes
Outcomes
Outline

Introduction
  Design
  Development
  Status

Teaching
  Courses
  Google Summer of Code
  Google Code-in

Research
  New language pairs
  Applying unsupervised methods
  Hybrid systems

Future work and challenges
  Challenges
  Plans
  Collaboration
New language pairs

Creating a new language pair involves:

- Describing the morphology as a finite-state transducer
- Writing a disambiguation grammar, or annotating a corpus
- Constructing a bilingual dictionary
- Developing a contrastive grammar of two languages

Many languages that Apertium works with have

- Limited grammatical descriptions
- Few or inexistent lexical resources
Unsupervised methods

« Learning statistical models from unlabelled data »

In Apertium:

- How to improve an existing MT system
- Remove a module from the pipeline and try and relearn it
- Applied to part-of-speech tagging and lexical selection

General method:

- Generate all possible outcomes
- Score the outcomes on a target-language model
- Use fractional counts as input into supervised algorithm
Unsupervised: Lexical selection

Monolingual corpus

Koivun syyt ovat yleensä suorat.
Se on sinun syytäsi.
Uniapnean syyt ja siihen vaikuttavat tekijät.
Ilmastonmuutoksen syyt ja seuraukset.
Onko tarkoitus tukkia puun syyt?
Petsi myös jättää puun syyt enemmän näkyviin.
### Unsupervised: Lexical selection

#### Expand possible translations

<table>
<thead>
<tr>
<th>Finnish</th>
<th>English</th>
</tr>
</thead>
<tbody>
<tr>
<td>Koivun syyt ovat yleensä suorat.</td>
<td>Birch grains are usually straight.</td>
</tr>
<tr>
<td>Birch reasons are usually straight.</td>
<td>Birch faults are usually straight.</td>
</tr>
<tr>
<td>It is your grain.</td>
<td></td>
</tr>
<tr>
<td>It is your reason.</td>
<td></td>
</tr>
<tr>
<td>It is your fault.</td>
<td></td>
</tr>
<tr>
<td>Se on sinun syytäsi.</td>
<td>Sleep apnea reasons and factors contributing to it.</td>
</tr>
<tr>
<td>It is your grain.</td>
<td>Sleep apnea grains and factors contributing to it.</td>
</tr>
<tr>
<td>It is your reason.</td>
<td>Sleep apnea faults and factors contributing to it.</td>
</tr>
<tr>
<td>It is your fault.</td>
<td></td>
</tr>
<tr>
<td>Uniapnean syyt ja siihen vaikuttavat tekijät.</td>
<td>Sleep apnea reasons and factors contributing to it.</td>
</tr>
<tr>
<td>Sleep apnea grains and factors contributing to it.</td>
<td>Sleep apnea faults and factors contributing to it.</td>
</tr>
<tr>
<td>Sleep apnea faults and factors contributing to it.</td>
<td></td>
</tr>
<tr>
<td>Ilmastonmuutoksen syyt ja seuraukset.</td>
<td>Global warming reasons and consequences.</td>
</tr>
<tr>
<td>Global warming grains and consequences.</td>
<td>Global warming faults and consequences.</td>
</tr>
<tr>
<td>There on surface is painted tree reasons become visible.</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted tree grains become visible.</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted tree faults become visible.</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted wood reasons become visible.</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted wood grains become visible.</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted wood faults become visible.</td>
<td></td>
</tr>
<tr>
<td>Siinä pintaan maalataan puun syyt näkyviin.</td>
<td>Stain also leaves tree reasons more visible.</td>
</tr>
<tr>
<td>There on surface is painted wood reasons become visible.</td>
<td>Stain also leaves tree grains more visible.</td>
</tr>
<tr>
<td>There on surface is painted wood faults become visible.</td>
<td>Stain also leaves tree faults more visible.</td>
</tr>
<tr>
<td>Stain also leaves wood reasons more visible.</td>
<td>Stain also leaves wood grains more visible.</td>
</tr>
<tr>
<td>Petsi myös jättää puun syyt enemmän näkyviin.</td>
<td>Stain also leaves wood faults more visible.</td>
</tr>
<tr>
<td>Stain also leaves tree reasons more visible.</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves tree grains more visible.</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves tree faults more visible.</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves wood reasons more visible.</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves wood grains more visible.</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves wood faults more visible.</td>
<td></td>
</tr>
</tbody>
</table>
### Unsupervised: Lexical selection

#### Score on language model and normalise

<table>
<thead>
<tr>
<th>Finnish</th>
<th>English</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Koivun syyt ovat yleensä suorat.</td>
<td>Birch grains are usually straight.</td>
<td>0.20</td>
</tr>
<tr>
<td>Birch reasons are usually straight.</td>
<td>0.36</td>
<td></td>
</tr>
<tr>
<td>Birch faults are usually straight.</td>
<td>0.44</td>
<td></td>
</tr>
<tr>
<td>It is your grain.</td>
<td></td>
<td>0.04</td>
</tr>
<tr>
<td>It is your reason.</td>
<td></td>
<td>0.12</td>
</tr>
<tr>
<td>It is your fault.</td>
<td></td>
<td>0.84</td>
</tr>
<tr>
<td>Sleep apnea reasons and factors contributing to it.</td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>Sleep apnea grains and factors contributing to it.</td>
<td>0.26</td>
<td></td>
</tr>
<tr>
<td>Sleep apnea faults and factors contributing to it.</td>
<td>0.19</td>
<td></td>
</tr>
<tr>
<td>Global warming reasons and consequences.</td>
<td>0.98</td>
<td></td>
</tr>
<tr>
<td>Global warming grains and consequences.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>Global warming faults and consequences.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted tree reasons become visible.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted tree grains become visible.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted tree faults become visible.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted wood reasons become visible.</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted wood grains become visible.</td>
<td>0.92</td>
<td></td>
</tr>
<tr>
<td>There on surface is painted wood faults become visible.</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves tree reasons more visible.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves tree grains more visible.</td>
<td>0.1</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves tree faults more visible.</td>
<td>0.32</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves wood reasons more visible.</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves wood grains more visible.</td>
<td>0.06</td>
<td></td>
</tr>
<tr>
<td>Stain also leaves wood faults more visible.</td>
<td>0.59</td>
<td></td>
</tr>
</tbody>
</table>

- **Uniapnean syyt ja siihen vaikuttavat tekijät.**
- **Ilmastonmuutoksen syyt ja seuraukset.**
- **Siinä pintaan maalataan puun syyt näkyviin.**
- **Petsi myös jättää puun syyt enemmän näkyviin.**
Unsupervised: Lexical selection

Extract \( n \)-grams and count

<table>
<thead>
<tr>
<th>( n )-gram</th>
<th>'reason'</th>
<th>'grain'</th>
<th>'fault'</th>
</tr>
</thead>
<tbody>
<tr>
<td>syyt</td>
<td>2.07</td>
<td>1.51</td>
<td>2.42</td>
</tr>
<tr>
<td>koivun syyt</td>
<td>0.36</td>
<td>0.20</td>
<td>0.44</td>
</tr>
<tr>
<td>syyt ovat</td>
<td>0.36</td>
<td>0.20</td>
<td>0.44</td>
</tr>
<tr>
<td>sinun syytäsi</td>
<td>0.12</td>
<td>0.04</td>
<td>0.84</td>
</tr>
<tr>
<td>uniapnean syyt</td>
<td>0.55</td>
<td>0.26</td>
<td>0.19</td>
</tr>
<tr>
<td>syyt ja</td>
<td>1.53</td>
<td>0.27</td>
<td>0.20</td>
</tr>
<tr>
<td>ilmastonmuutoksen syyt</td>
<td>0.98</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>puun syyt</td>
<td>0.06</td>
<td>1.00</td>
<td>0.94</td>
</tr>
<tr>
<td>syyt näkyviin</td>
<td>0.04</td>
<td>0.93</td>
<td>0.03</td>
</tr>
<tr>
<td>syyt enemmän</td>
<td>0.02</td>
<td>0.07</td>
<td>0.91</td>
</tr>
</tbody>
</table>
Unsupervised: Lexical selection

What then?

- Take counts and feed into ML-algorithm of choice
  - For example MaxEnt, as in my thesis
- Get the same quality as TL-model, using only SL-information
- No need to make multiple translations at runtime!
- Learn translation probabilities without a parallel corpus!
Hybrid systems

Using rule-based systems inside SMT:

- Many approaches:
  - Synthetic data (either wordforms or phrases)
  - Incorporating linguistic information (morphology or syntax)
- A system with data from Apertium came second in WMT

Using statistics in rule-based systems:

- Most rule-based systems already have statistics to some extent
Outline

Introduction
  Design
  Development
  Status

Teaching
  Courses
  Google Summer of Code
  Google Code-in

Research
  New language pairs
  Applying unsupervised methods
  Hybrid systems

Future work and challenges
  Challenges
  Plans
  Collaboration
Major challenges

- Coverage
- Finding and motivating users
Coverage

Given adequate (> 95%) coverage, we are competitive with Google translate:

- Evaluations of Slovenian→Serbo-Croatian, Afrikaans→Dutch, Swedish→Danish, Danish→Norwegian, Maltese→Arabic have shown this

However, most pairs are prototypes:

- Coverage around 80%–85%
- Increasing lexical coverage beyond this is pretty boring
- Difficult to motivate people
Finding users

Challenge:
- Apertium often finds itself translating in ‘non-canonical’ translation directions
- People are fond of translating from English (or French)

Successes:
- Spanish–Portuguese (Prompsit + AutoDesk)
- Romance languages in general

“to do”:
- Turkic languages
- Slavic languages
- Uralic languages
Factoring out language independent resources

- Apertium began with three language pairs, now has thirty
- Most language pairs developed by *copying* previous data
- Result → 13 copies of Spanish morphological dictionary
- This year we started to separate out monolingual data
- Now many languages have their own language directory
- Pairs then depend on a single monolingual source
Factoring out language independent resources

- Minor tagset differences that were introduced
- Classification differences
- “Multiwords”

<table>
<thead>
<tr>
<th>Pair</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>spa-cat</td>
<td>el&lt;det&gt;&lt;def&gt;&lt;f&gt;&lt;sg&gt; mismo&lt;adj&gt;&lt;f&gt;&lt;sg&gt; casa&lt;n&gt;&lt;f&gt;&lt;sg&gt;</td>
</tr>
<tr>
<td>spa-eng</td>
<td>el mismo&lt;det&gt;&lt;ind&gt;&lt;f&gt;&lt;sg&gt; casa&lt;n&gt;&lt;f&gt;&lt;sg&gt;</td>
</tr>
</tbody>
</table>

- Problem #1: Generation errors
- Problem #2: Solving these errors will not solve any problems
Tree-based transfer

Languages with radically different word order could benefit from:
- Long distance reordering and agreement
- “recursive” transfer rules

Current prototype inspired by METAL and MorphoLogic:
- Bison (CFG) based transformer
- Build target-language tree as you parse
- Pattern-Action, like current Apertium

\[ \text{Ssub} \rightarrow \text{that SV \{ $2 \ $1 \}} \]
I know that you went to Kazakhstan.

Мен сіздің Қазақстанға барғаныңызды білемін.

Minä sinun Kazakstaniin menneen tiedän.
Weighted pipeline

Apertium includes probabilistic components in the pipeline:
- Part-of-speech tagger calculates tag probabilities
- Lexical-selection module allows for translation probabilities

However:
- No probabilistic information encoded in the lexicon
- Rules are not weighted

In general:
- We should be able to take into account weights at all stages
- We should never output something less probable
- But rules should always allow us to control the output
  - And output should be predictable!
Lexicon graph

- There are 40 ‘released’ bilingual dictionaries in Apertium
- We have a program that ‘crosses’ two dictionaries, e.g.
  - Spanish-French + Spanish-Occitan → French-Occitan

Questions:
- What happens with ambiguities?
- Can all other dictionaries help?
Lexicon graph

piim

milk

maito
Lexicon graph

piim
\[\text{est-eng}\]
milk
\[\text{eng-fin}\]
maito

piim
\[\text{est-eng}\]
milk
\[\text{eng-fin}\]
maito
\[\text{fin-est}\]
Lexicon graph
Lexicon graph
This is fairly novel, but similar work has been done before:


Lexicon graph
Constructions

- Some things don’t translate well if you translate word-by-word!

- The meaning is not so much in the “content” words, but in the expression.
Some things don’t translate well if you translate word-by-word!

Ei oo kaikki muumit laaksossa

The meaning is not so much in the “content” words, but in the expression.
Some things don’t translate well if you translate word-by-word!

Ei oo kaikki muumit laaksossa
Not all Moomin’s are in the valley

The meaning is not so much in the “content” words, but in the expression.
Constructions

▶ Some things don’t translate well if you translate word-by-word!

Ei oo kaikki muumit laaksossa
Not all Moomin’s are in the valley

A few sandwiches short of a picnic

▶ The meaning is not so much in the “content” words, but in the expression.
Some things don’t translate well if you translate word-by-word!

Ei oo kaikki muumit laaksossa
Not all Moomins are in the valley

A few sandwiches short of a picnic

Ei oo kaikki X:t Y:ssa → A few X short of a Y.
« A few Moomins short of a valley »

The meaning is not so much in the “content” words, but in the expression.
Software-engineering practices

With large software projects, following good engineering principles is often a challenge

- When Apertium was only pair-based this was easier. Each pair was an island.
- Now we have separated out monolingual data, changes in one place can have knock-on effects in other places.
- We have no full test suite that developers can rely on.
Language pairs

More language pairs!

- Almost there
  - Russian–Ukrainian
  - Kazakh–Kyrgyz
  - Kazakh–Karakalpak
  - Tatar→Russian

- Funded: North Sámi→Inari Sámi
Developers working on rule-based systems should collaborate!

- **Morphological descriptions**
  - Lemma lists categorised by morphological paradigm
- **Dictionaries**
  - Bi-/ multi-lingual correspondences between lemma + POS
- **Ideas**
  - Anything you can think of!
Giitu · Takk · Kiitos · Tack!