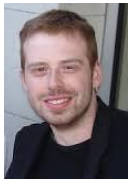
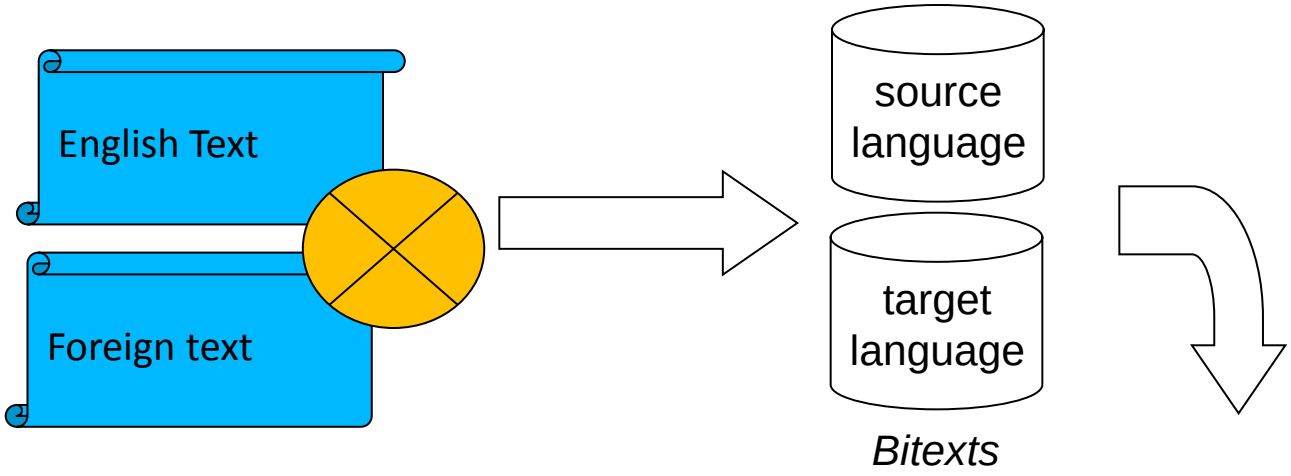
Lecture 6.4: MT framework

Some further discussion on how to prepare your texts for MT, launch Moses and evaluate results

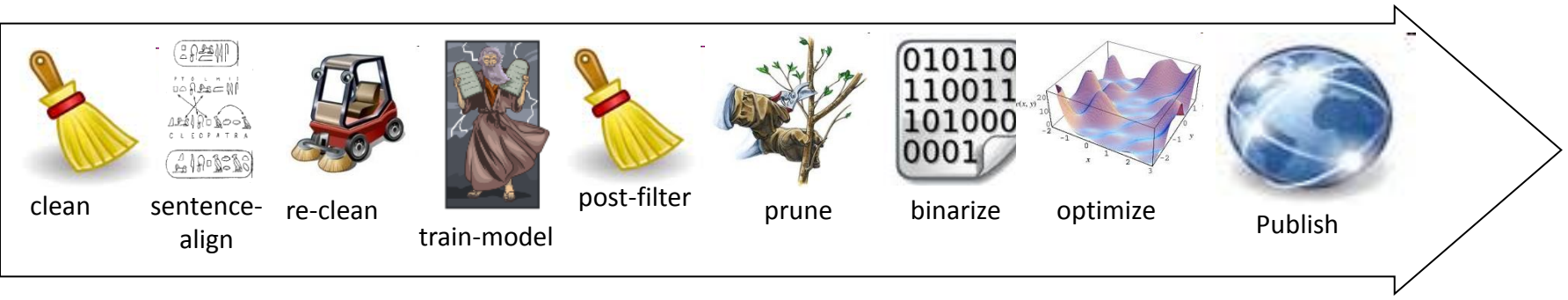
Bruno Pouliquen
+ Christophe Mazenc
+ Marcin Junczys-Dowmunt



MT building framework



The process prepares the raw texts for Moses, applies some post-processing (filter, pruning, binarization, optimization…) and offers various interfaces to translate



- Clean your documents
- Convert it to text (utf8)
- Tokenize
- Split into sentences
- Sentence align
- and clean again!

Clean documents

- Avoid noise!
- Language guesser: check that your English documents are not in another language
- Try to avoid mix-language documents
- Sometimes annexes are not translated
- Convert them to raw text
- Remove duplicate

Sentence alignment: tools

- Sentence length based aligner:

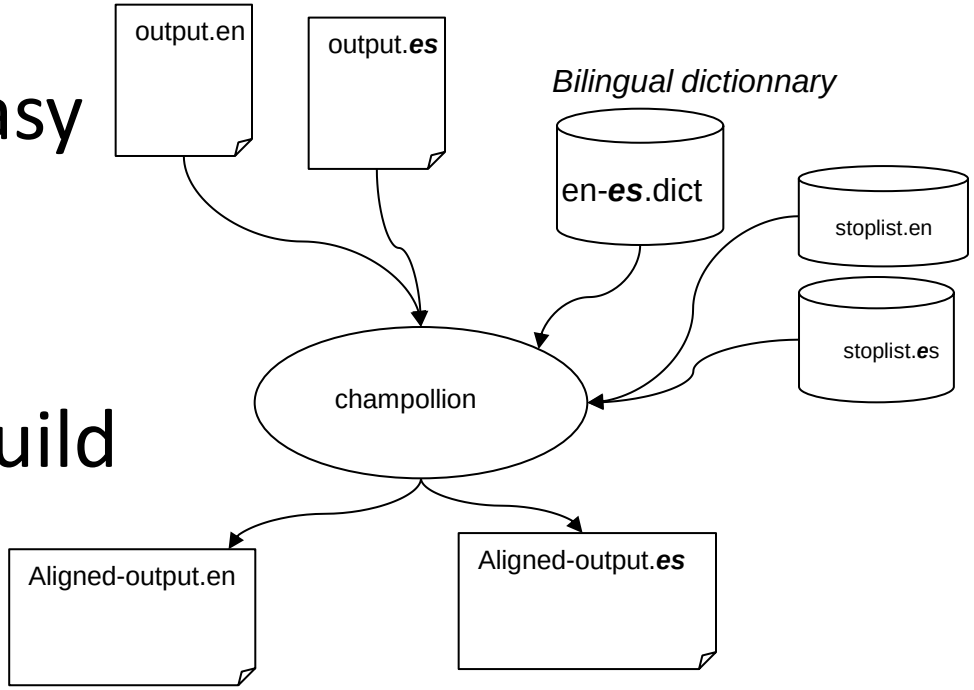
  Vanilla [Danielsson et al. 1994] based on [Gale&Church1991]
- Based on lexical information:
  - hunalign: https://github.com/danielvarga/hunalign
  - Champollion:http://sourceforge.net/projects/champollion/
- Based on MT and optimizing BLEU:

  Bleu align: https://github.com/rsennrich/bleualign

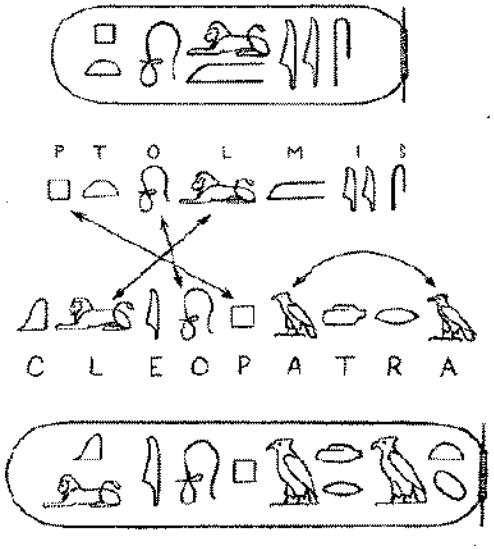
As we have to filter out long sentences, maybe we should split long sentences at this level

From raw text aligned documents to bitexts…

- Split documents into sentences
- E.g. Champollion
- Needs stopwords (easy to find)
- Need a bilingual dictionary (easy to build from Moses)



Champollion



- Input dictionaries:
  - bilingual dictionary (source <> destination)
  - English stoplist “meaningless” words in English
  - Foreign stoplist
- Outputs sentences with alignment (0-1,1-1,1-2,2-1 etc.)

Bilingual dictionary building...

- Problem:
  - No input bilingual dictionary
  - Out of domain dictionary
  - Outdated dictionary etc.
- Solution: create dictionary from the data
  - Out of a previously Moses trained model (just after the Giza alignment)
  - Work on the lex.e2f / lex.f2e

Example of post-aligned filters

- We now have sentences
- Filter out sentences badly aligned
- Filter out sentences loosely aligned with neighbors badly aligned

Clean long lines

- Filter out sentences having more than 40 words (what threshold to use?)
- Ratio (# src words/ # dst words) more than 9

Word alignment (giza) complexity is linear according to the number of sentences, however it is quadratic according to the number of tokens per sentence

Longer sentences=>CPU time may suffer, loose alignments

Shorted sentences=>Better for GIZA, but you may loose big part of your data

Language model

- Try to use lot of data…

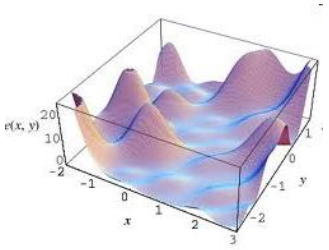
  But recent / clean and in-domain
- The number of n-grams depends on your memory (as usual find the right balance between efficiency, speed and space)
- Do not forget to binarize your “arpa” file

  kenlm/build_binary filename.arpa filename.binary
- See http://www.statmt.org/moses/?n=FactoredTraining.BuildingLanguageModel#ntoc20

Pruning

For big models, it is usually a good idea to “prune” the phrase table
See “discarding phrase table entries which do not have a good significance scores” (Johnson et al. 2007)

Parameter optimization (MERT)



Tune the weights that « maximize » BLEU scores on devset, in other words that minimize a given error measure (MERT=minimum error rate training)

See documentation on
http://www.statmt.org/moses/?n=FactoredTraining.Tuning

Binarization

A big model cannot stand in memory, solution: binarize the phrase table

Phrase table:
http://www.statmt.org/moses/?n=Advanced.RuleTables#ntoc3

Developed by my colleague Marcin Junczys-Dowmunt, huge compression!

```
~/mosesdecoder/bin/processPhraseTableMin -in ~/phrase-table.gz -out phrase-table
```

A side effect, you can then query directly the phrase table:

```
 ~/mosesdecoder/bin/queryPhraseTableMin [-a] -t model/phrase-table.minphr
Eg.
echo "green economy" | ~mosesdecoder/bin/queryPhraseTableMin -a -t ~/model/phrase-
    table.minphr
```

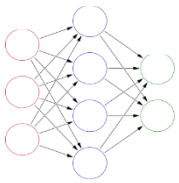
Training and scalability at WIPO
Size reduction zh-en

	Phrase table 0-0		Phrase table 0,1-0		Reordering model		Language model	
	M rows	Gb	M rows	Gb	M rows	Gb	M ngrams	Gb
Basic	806	100	974	130	806	89	584	23
Pruned	551	69	623	83	551	61	388	16
Binarized		6.4		7.4		4.2		4.6

342Gb

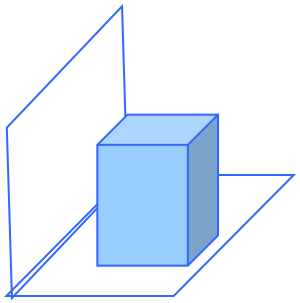
22.6Gb (6.6%)

0.5 Gb



New Neural MT zh-en WIPO model…

Evaluation

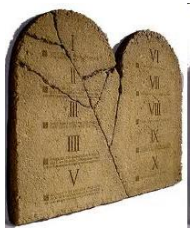


- Always question your bleu score!
- Always check if your testset is representative
- As usual, always look at your data

Trainset / devset / testset

- Trainset: Set of sentences used by your machine to learn (& language model generation)
- Devset: Set of sentences used to optimize the model parameters (Moses “MERT”)
- Testset: Set of sentences used for comparison between automatic translation and human translation (e.g. BLEU)

Trainset / devset / testset
few commandments



- Never mix trainset and testset
  (Even for the language model)
- Always test on held-out documents
- Try to have a representative test/dev set
  (In some context: translate new documents)
  - Test on newest documents
  - Develop on newest documents
- Use appropriate evaluation measure (human evaluation, BLEU etc.), see lecture 6

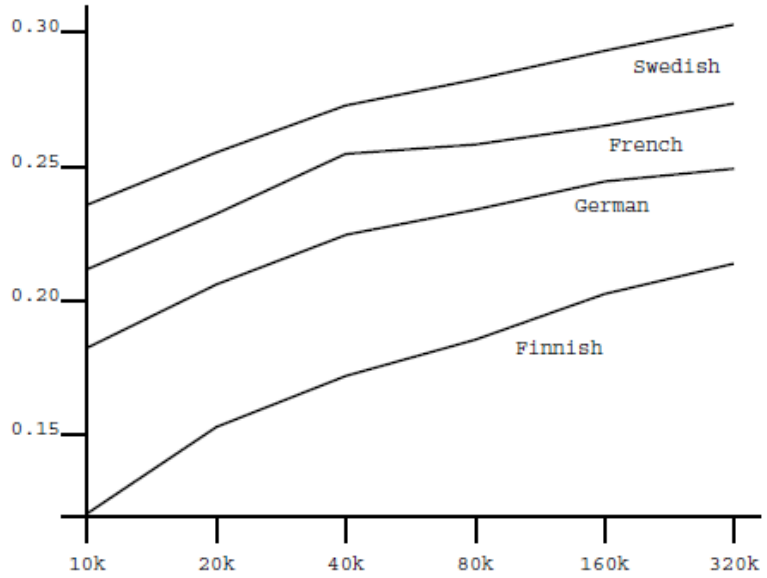
Training data can be huge

Example in WIPO

	M Segments	Mwords
PT:	7	209
JA:	112	4000
ZH:	62	1872
FR:	18	488
De:	40	1455
Ru:	13	386
KO:	44	1481

SMT : how well does it work ?

More Data, Better Translations



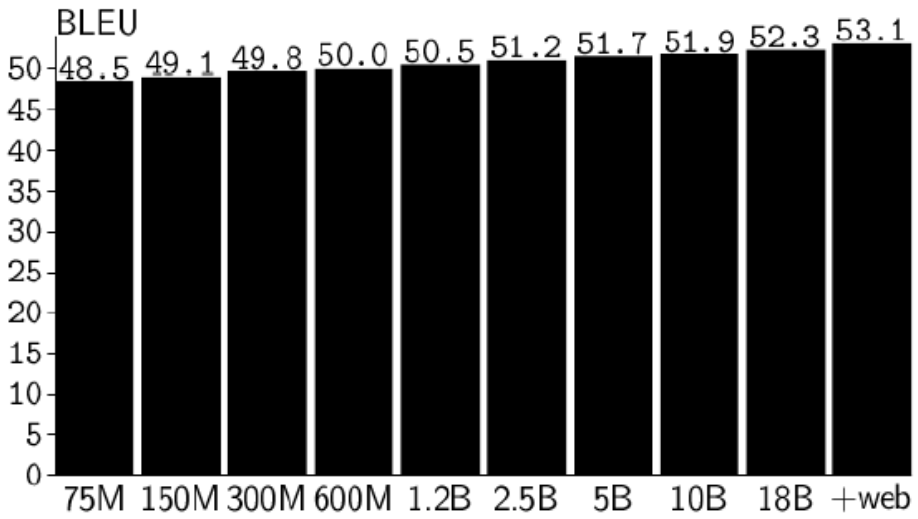
[from Koehn, 2003: Europarl]

- Log-scale improvements on BLEU:
  Doubling the training data gives constant improvement (+1 %BLEU)

Credit: P. Koehn

SMT : how well does it work ?

More LM Data, Better Translations

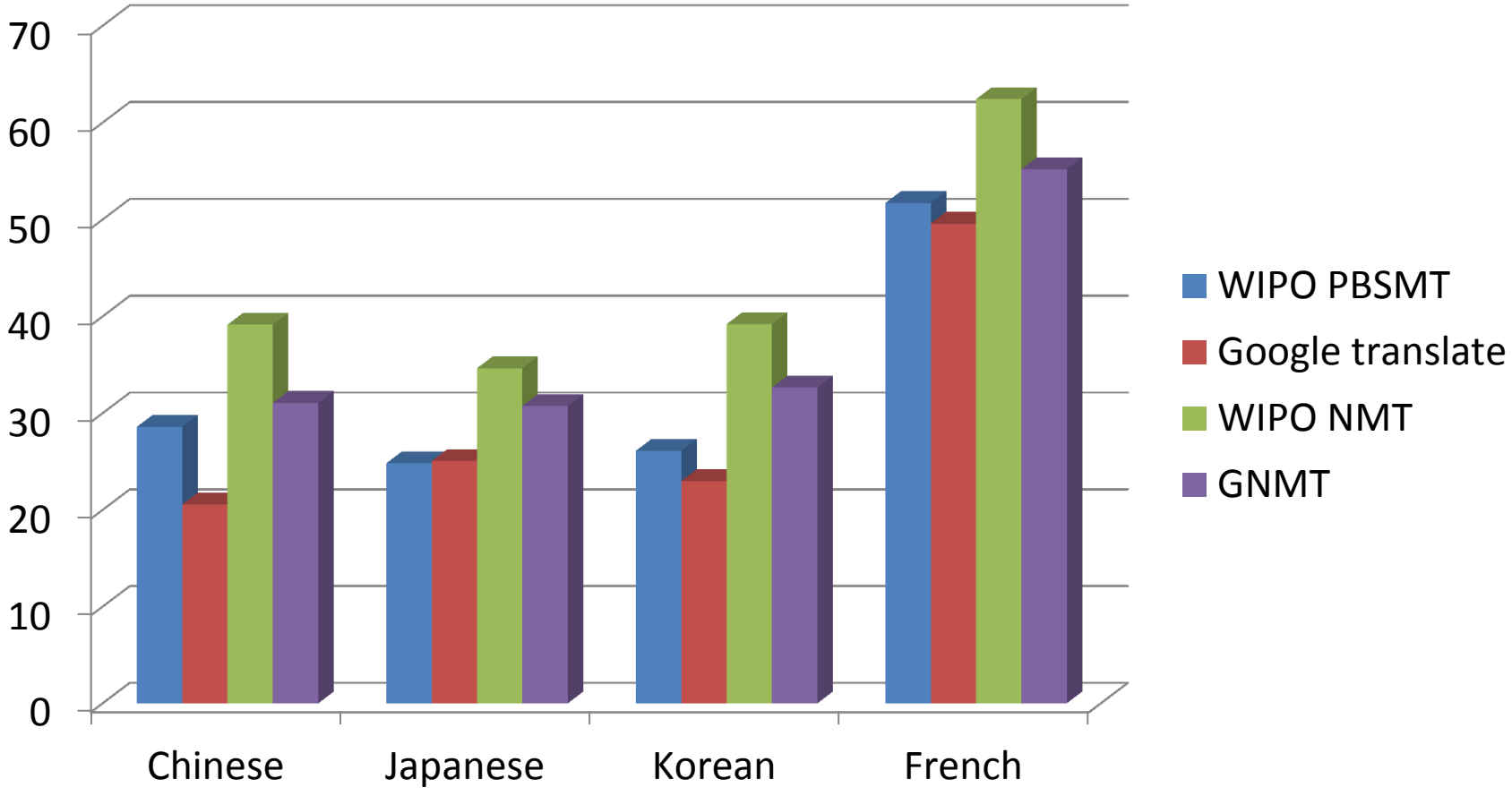


[from Och, 2005: MT Eval presentation]

- Also log-scale improvements on BLEU:
  doubling the training data gives constant improvement (+0.5 %BLEU)
  (last addition is 218 billion words out-of-domain web data)

Credit: P. Koehn

WIPO Experiments



User acceptance

How WIPO translate is perceived among translators?

- When seen as a “translation accelerator”: useful
- When seen as “replacement for translator”: useless

- When proposed as a copy-paste tool: not used
- When integrated in translator’s environment: used daily

Discussion

- Try to build your own MT system!

Thank you for your attention

- شكرا لكم على اهتمامكم
- Merci pour votre attention!
- 感谢您的关注
- Grazie per la vostra attenzione!
- ¡ Gracias por su atención !
- Vielen Dank für Ihre Aufmerksamkeit!
- Obrigado pela vossa atenção!
- Dziękuję bardzo za Państwa uwagę!
- Děkujeme za Vaši pozornost!
- Ďakujem ti veľmi pekne za tvoju pozornosť
- Tänan tähelepanu eest!
- Благодарим за Вашето внимание!
- Tak for Jeres opmærksomhed!
- आप अपना ध्यान के लिए धन्यवाद
- Thank you for your attention!