lium

Introduction aux systèmes de traduction basés sur les séquences de mots hiérarchiques

Les fondements de Joshua, système de traduction open source

Yannick Estève - LIUM - Université du Maine

9 mars 2010

Sources

- **Hierarchical Phrase-Based Translation**, David Chiang, 2007, *Computational Linguistics 33(2):201-228*
- **Joshua: An Open Source Toolkit for Parsing-based Machine Translation**, Zhifei Li, Chris Callison-Burch, Chris Dyer, Juri Ganitkevitch, Sanjeev Khudanpur, Lane Schwartz, Wren N. G. Thornton, Jonathan Weese and Omar F. Zaidan, *WMT 2009*
- **http://cs.jhu.edu/~ccb/joshua/**
  - HOW-TO GUIDE: Installing and running the Joshua Decoder by Chris Callison-Burch (Released: June 12, 2009)
- **http://www.clsp.jhu.edu/wiki2/Joshua_Lab**

Traduction automatique à base de séquence de mots

- Les systèmes de traduction automatique probabilistes performants actuels sont généralement des systèmes basés sur la traduction par séquences de mots
  - ex : Moses qui utilisent les alignements de Giza++

Limite de l'approche basée sur les séquences de mots

澳洲	是	与	北韩	有	邦交	的	少数	国家	之一	。
Aozhou	shi	yu	Beihan	you	bangjiao	de	shaoshu	guojia	zhiyi	.
Australia	is	with	North Korea	have	dipl. rels.	that	few	countries	one of	.

Australia is one of the few countries that have diplomatic relations with North Korea.

Le ré-ordonnancement des mots pose encore des problèmes, même si l'approche par séquences de mots permet de bien modéliser les événements observés dans le corpus d'apprentissage

[Aozhou] [shi]$_1$ [yu Beihan]$_2$ [you] [bangjiao] [de shaoshu guojia zhiyi] [.]

[Australia] [has] [dipl. rels.] [with North Korea]$_2$ [is]$_1$ [one of the few countries] [.]

Les séquences de mots hiérarchiques : motivations (1)

- L'approche par séquences de mots modélise correctement le réordonnancement de certains mots (séquence par séquence)
- Pour mieux modéliser le réordonnancement, on souhaite utiliser des séquences de séquences : ce sont les séquences de mots hiérarchiques
    - permet de généraliser des observations

Les séquences de mots hiérarchiques : motivations (2)

澳洲	是	与	北韩	有	邦交	的	少数	国家	之一	。
Aozhou	shi	yu	Beihan	you	bangjiao	de	shaoshu	guojia	zhiyi	.
Australia	is	with	North Korea	have	dipl. rels.	that	few	countries	one of	.

Australia is one of the few countries that have diplomatic relations with North Korea.

[Aozhou] [shi]$_1$ [yu Beihan]$_2$ [you] [bangjiao] [de shaoshu guojia zhiyi] [.]

[Australia] [has] [dipl. rels.] [with North Korea]$_2$ [is]$_1$ [one of the few countries] [.]

- soit la séquence de mots hiérarchique suivante :

  ⟨yu [1] you [2], have [2] with [1]⟩

➡ [1] et [2] sont des emplacements pour des sous-séquences qui correspondent : cette approche est plus puissante que l'approche par séquences de mots conventionelles

Les grammaires hors contexte synchrones : principes (1)

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle$$

X est un non terminal,

γ et α sont des séquences d'éléments terminaux et non terminaux,

$\sim$ est fonction bijective entre éléments non terminaux de γ et éléments non terminaux de α.

Les grammaires hors contexte synchrones : principes (2)

Séquences de mots hiérarchiques :

$$X \rightarrow \langle \text{yu } X_{[1]} \text{ you } X_{[2]}, \text{have } X_{[2]} \text{ with } X_{[1]} \rangle \quad (6)$$

$$X \rightarrow \langle X_{[1]} \text{ de } X_{[2]}, \text{the } X_{[2]} \text{ that } X_{[1]} \rangle \quad (7)$$

$$X \rightarrow \langle X_{[1]} \text{ zhiyi, one of } X_{[1]} \rangle \quad (8)$$

Séquences de mots conventionnelles :

$$X \rightarrow \langle \text{Aozhou, Australia} \rangle \quad (9)$$

$$X \rightarrow \langle \text{Beihan, North Korea} \rangle \quad (10)$$

$$X \rightarrow \langle \text{shi, is} \rangle \quad (11)$$

$$X \rightarrow \langle \text{bangjiao, diplomatic relations} \rangle \quad (12)$$

$$X \rightarrow \langle \text{shaoshu guojia, few countries} \rangle \quad (13)$$

Deux règles primordiales :

$$S \rightarrow \langle S_{[1]} X_{[2]}, S_{[1]} X_{[2]} \rangle \quad (14)$$

$$S \rightarrow \langle X_{[1]}, X_{[1]} \rangle \quad (15)$$

Les grammaires hors contexte synchrones : exemple de dérivation

$\langle S_{\boxed{1}}, S_{\boxed{1}} \rangle$

$\overset{(14)}{\Rightarrow} \langle S_{\boxed{2}} X_{\boxed{3}}, S_{\boxed{2}} X_{\boxed{3}} \rangle$

$\overset{(14)}{\Rightarrow} \langle S_{\boxed{4}} X_{\boxed{5}} X_{\boxed{3}}, S_{\boxed{4}} X_{\boxed{5}} X_{\boxed{3}} \rangle$

$\overset{(15)}{\Rightarrow} \langle X_{\boxed{6}} X_{\boxed{5}} X_{\boxed{3}}, X_{\boxed{6}} X_{\boxed{5}} X_{\boxed{3}} \rangle$

$\overset{(9)}{\Rightarrow} \langle$ Aozhou $X_{\boxed{5}} X_{\boxed{3}}$, Australia $X_{\boxed{5}} X_{\boxed{3}} \rangle$

$\overset{(11)}{\Rightarrow} \langle$ Aozhou shi $X_{\boxed{3}}$, Australia is $X_{\boxed{3}} \rangle$

$\overset{(8)}{\Rightarrow} \langle$ Aozhou shi $X_{\boxed{7}}$ zhiyi, Australia is one of $X_{\boxed{7}} \rangle$

$\overset{(7)}{\Rightarrow} \langle$ Aozhou shi $X_{\boxed{8}}$ de $X_{\boxed{9}}$ zhiyi, Australia is one of the $X_{\boxed{9}}$ that $X_{\boxed{8}} \rangle$

$\overset{(6)}{\Rightarrow} \langle$ Aozhou shi yu $X_{\boxed{1}}$ you $X_{\boxed{2}}$ de $X_{\boxed{9}}$ zhiyi,
Australia is one of the $X_{\boxed{9}}$ that have $X_{\boxed{2}}$ with $X_{\boxed{1}} \rangle$

$\overset{(10)}{\Rightarrow} \langle$ Aozhou shi yu Beihan you $X_{\boxed{2}}$ de $X_{\boxed{9}}$ zhiyi,
Australia is one of the $X_{\boxed{9}}$ that have $X_{\boxed{2}}$ with North Korea$\rangle$

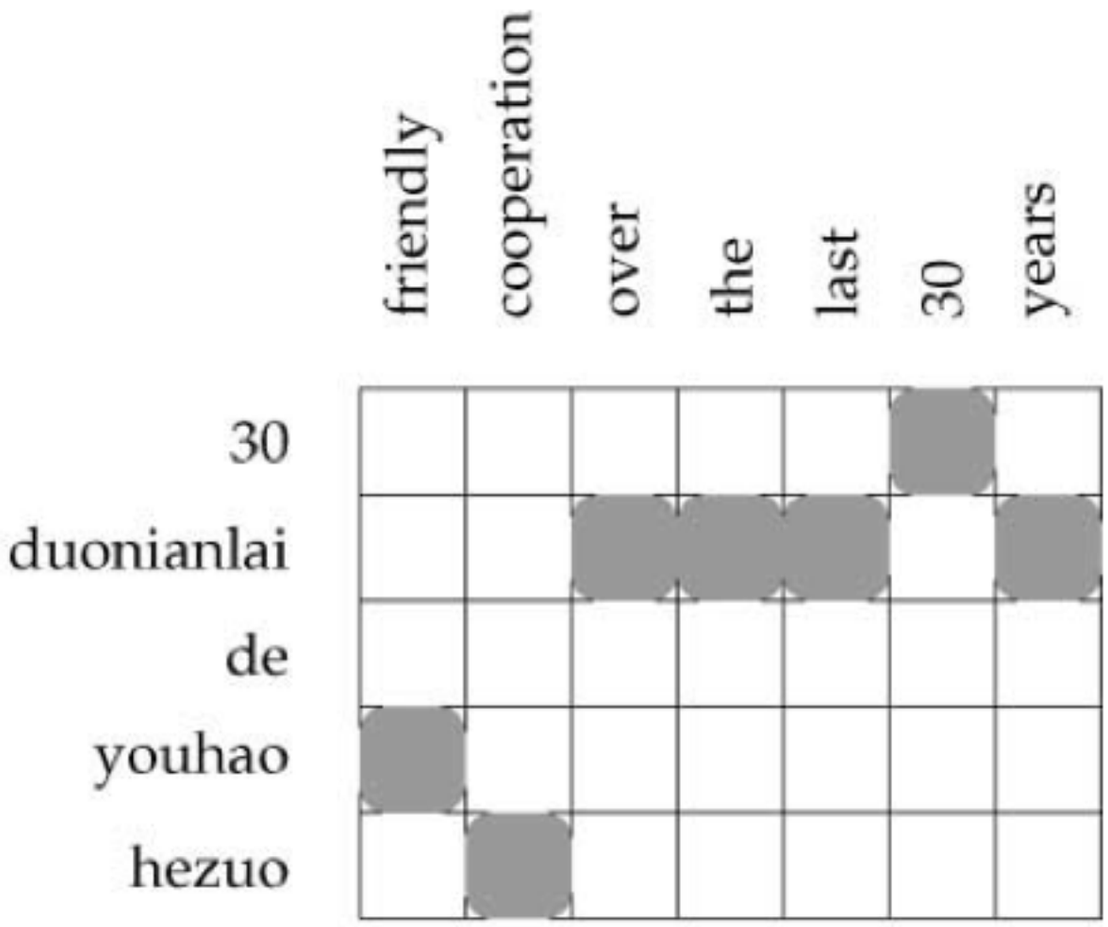
$\overset{(12)}{\Rightarrow} \langle$ Aozhou shi yu Beihan you bangjiao de $X_{\boxed{9}}$ zhiyi,
Australia is one of the $X_{\boxed{9}}$ that have diplomatic relations with North Korea$\rangle$

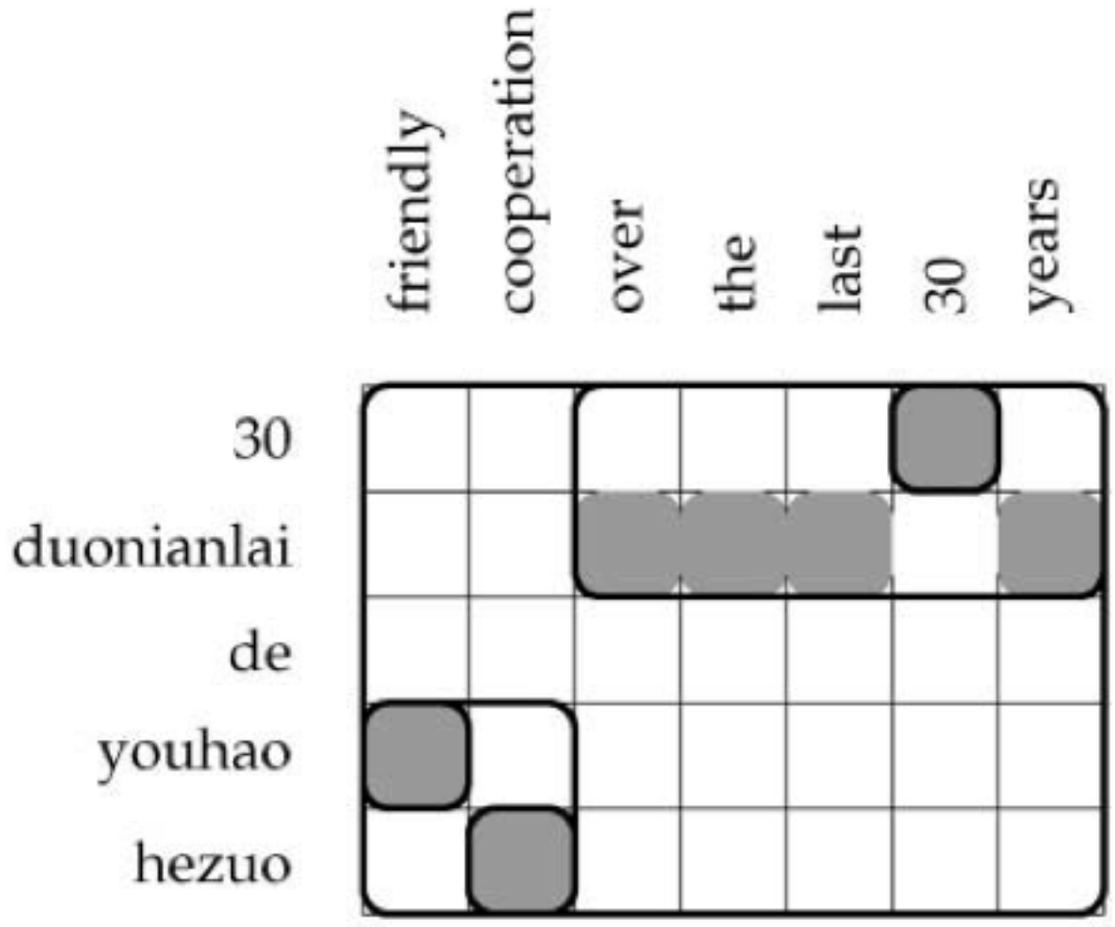
$\overset{(13)}{\Rightarrow} \langle$ Aozhou shi yu Beihan you bangjiao de shaoshu guojia zhiyi,
Australia is one of the few countries that have diplomatic relations with North Korea$\rangle$

Les grammaires hors contexte synchrones : apprentissage (1)

- Extraction des règles sans annotation syntaxique
- Utilisation d'alignements mot par mot (Giza++ ou autre)
- Etape 1 : définition des séquences de mots initiales (conventionnelles) comme "d'habitude"

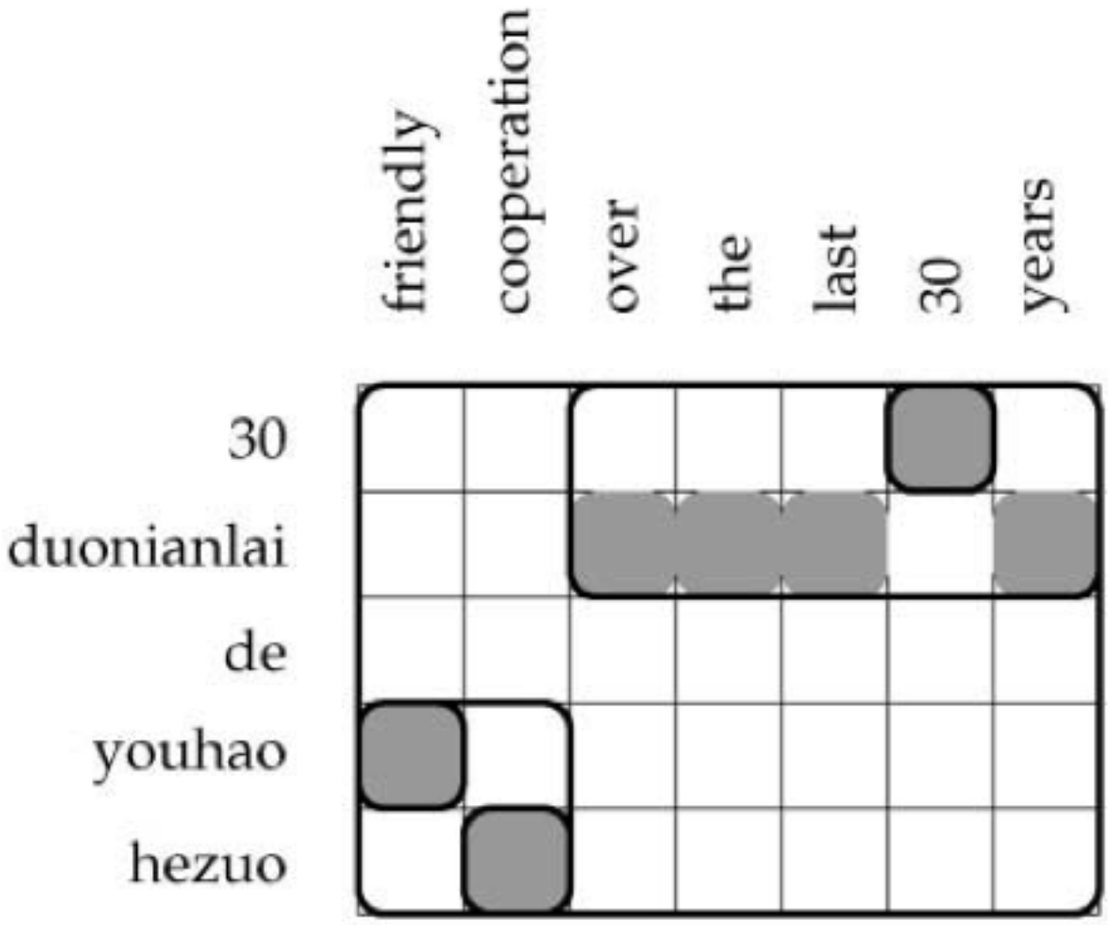
Les grammaires hors contexte synchrones : apprentissage (2)

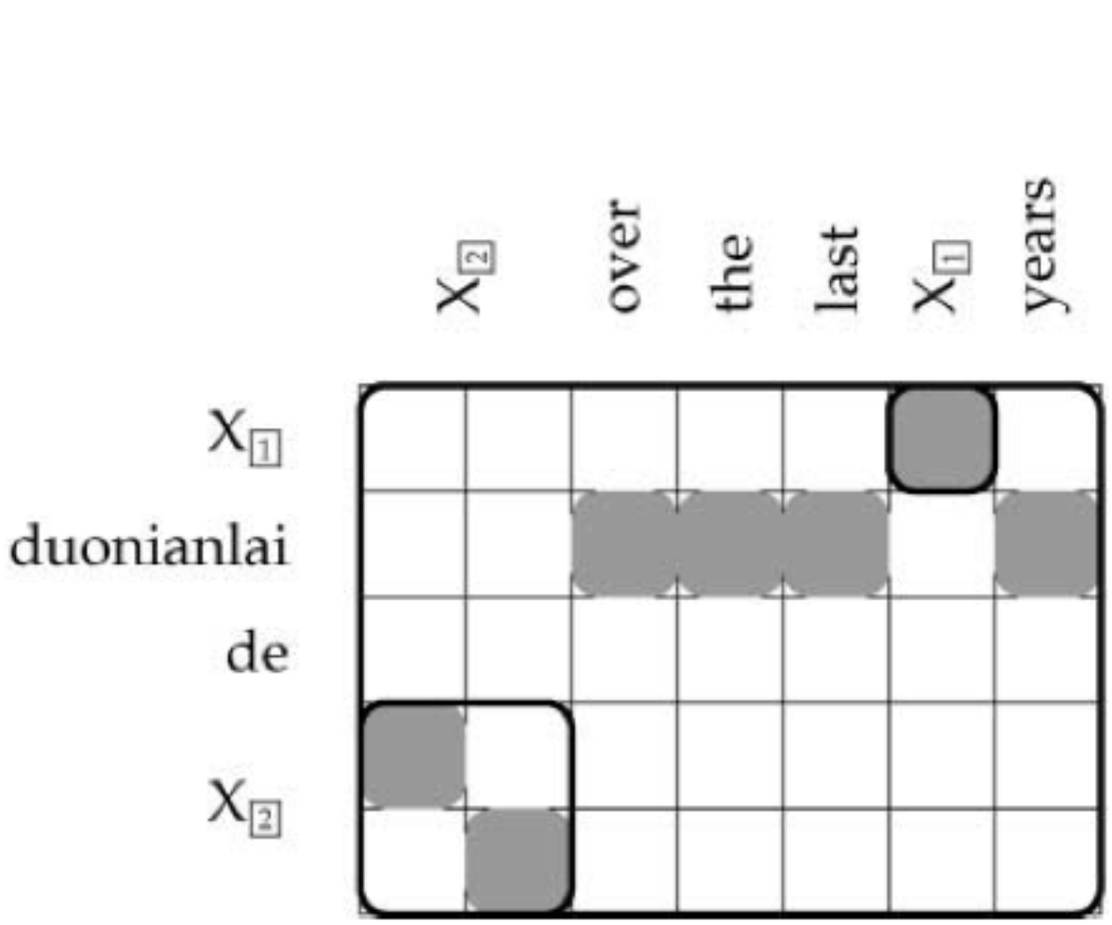




Il doit y avoir au moins un mot d'une séquence aligné avec un mot dans l'autre séquence, et aucun mot dans une séquence doit être aligné avec un mot en dehors de l'autre séquence

Les grammaires hors contexte synchrones : apprentissage (3)





On cherche ensuite des séquences de mots qui sont contenues dans d'autres séquences de mots et on les remplaces par des non terminaux

Les grammaires hors contexte synchrones : apprentissage (4)

- Afin de limiter le nombre de règles extraites, des heuristiques sont appliquées
  - ex 1: comme pour l'approche par séquences conventionnelles, les séquences initiales sont limités à 10 mots
  - ex 2 : dans une règle de grammaire, le nombre de non terminaux et terminaux de la langue source est limité à 5
  - ex 3 : les non terminaux adjacents dans la langue source sont interdits
  - ...

Les grammaires hors contexte synchrones : autres règles

- *Glue rules*
  - Règles de production de départ

$$S \rightarrow \langle S_{[1]}\ X_{[2]},\ S_{[1]}\ X_{[2]} \rangle \quad (14)$$

$$S \rightarrow \langle X_{[1]},\ X_{[1]} \rangle \quad (15)$$

  - Permet de représenter une phrase comme une séquence de chunk et de la traduire chunk par chunk
- *Entity rules*
  - Règles permettant de modéliser des dates, des nombres
  - Peuvent être créées manuellement

Les grammaires hors contexte synchrones et probabilités

- Afin de pouvoir prendre en compte des probabilités lors de l'utilisation d'une grammaires hors contexte synchrones, les règles de production sont pondérées
    - Classique, utilisé avec les grammaires probabilistes

Les grammaires hors contexte synchrones et probabilités

- Les poids des règles sont combinés à d'autres scores :
  - modèles n-gram
  - $P(\gamma \mid \alpha)$ et $P(\alpha \mid \gamma)$
  - $P_w(\gamma \mid \alpha)$ et $P_w(\alpha \mid \gamma)$ les poids lexicaux qui estiment la qualité de la traduction des mots de α dans γ (et vice-versa)
  - une pénalité pour chaque utilisation de règle afin de choisir entre de courtes ou de longues dérivations

Les grammaires hors contexte synchrones et probabilités

- Pour combiner l'ensemble de ces scores, on passe par une phase d'optimisation par 'minimization error rate' (ou maximisation du score BLEU ...)
- Classique aussi

Système de traduction hiérarchique : principes de décodage

- Utilisation de l'algorithme CYK : Cock-Younger-Kasami
  - En fait algo. CYK modifié qui ne passe pas par les formes normales de Chomsky (car limitations imposées de l'extraction des règles)
- Recherche de la phrase de probabilité maximale
  - En fait, sous-optimale en raison d'heuristiques d'élagage (algo CYK en $o(n^3)$)

La pratique avec Joshua : présentation de Joshua

- Joshua est un décodeur implémentant l'approche par séquences hiérarchiques
- Il est accompagné de l'ensemble des outils nécessaire à son fonctionnement : apprentissage des règles, décodeur, optimisation des poids, minimisation d'erreur, ...
- Il est totalement écrit en Java (approche modulaire)
- Open source (LGPL)

La pratique avec Joshua : alignement

- Même préparation qu'habituellement (normalisation des textes, tokenization, etc.)
  - pas besoin d'annotation syntaxique
- Alignement de mots :
  - Utilisation de Giza++ possible (nb : pas la peine d'utiliser toute la séquence de traitements)
  - Dans mes expés : utilisation de berkeleyAligner

La pratique avec Joshua : Apprentissage (1)

- Compilation du corpus d'apprentissage

```
#Compilation des données, Joshua
java -Xmx1024m -cp bin joshua.corpus.suffix_array.Compile \
                      /raid2/yannick/iwslt09/train/btec.ar \
                      /raid2/yannick/iwslt09/train/btec.en \
                      /raid2/yannick/iwslt09/output.berkeleyAligner/
training.ar-en.align \
                      /raid2/yannick/iwslt09/output.josh
```

La pratique avec Joshua : Apprentissage (2)

- Extraction des règles pour un ensemble de données fixé (ici alldev.ar)
  - limite le nombre de règles

```
#Extraction des règles de grammaires, Joshua (version 1.1)
java -Dfile.encoding=UTF8 -Xmx1g -cp bin joshua.prefix_tree.ExtractRules \
                                        --binary-source --binary-target \
              --source=/raid2/yannick/iwslt09/output.josh/source.corpus \
              --target=/raid2/yannick/iwslt09/output.josh/target.corpus \
          --source-vocab=/raid2/yannick/iwslt09/output.josh/common.vocab \
          --target-vocab=/raid2/yannick/iwslt09/output.josh/common.vocab \
    --source-suffixes=/raid2/yannick/iwslt09/output.josh/source.suffixes \
    --target-suffixes=/raid2/yannick/iwslt09/output.josh/target.suffixes \
        --alignments=/raid2/yannick/iwslt09/output.josh/alignment.grids \
                            --alignmentsType=MemoryMappedAlignmentGrids \
                            --test=/raid2/yannick/iwslt09/dev/alldev.ar \
              --output=/raid2/yannick/iwslt09/output.josh/alldev09.ar-en.grammar.raw
```

La pratique avec Joshua : décodage

- Utilisation d'un modèle n-gram estimé par ailleurs (SRILM par exemple)
- Pour configurer le décodeur Joshua, il faut éditer un fichier:

```
lm_file=/raid2/yannick/iwslt09/lm/btecd123.dev6.iw09b.4g.kn-int.sblm

tm_file=/raid2/yannick/iwslt09/output.josh/dev6.ar-en.grammar.raw
tm_format=hiero

glue_file=/raid2/yannick/iwslt09/output.josh/hiero.glue
glue_format=hiero

#lm config
...
order=4
###### model weights
#lm order weight
lm 1.0
... tm_config, pruning_config, nbest_config, ...
#phrasemodel owner column(0-indexed) weight
phrasemodel pt 0 1.066893
phrasemodel pt 1 0.752247
phrasemodel pt 2 0.589793

#arityphrasepenalty owner start_arity end_arity weight
#arityphrasepenalty pt 0 0 1.0
#arityphrasepenalty pt 1 2 -1.0

#phrasemodel mono 0 0.5

#wordpenalty weight
wordpenalty -2.844814
```

La pratique avec Joshua :
La pratique avec Joshua : ZMERT

- Optimization des poids (voir fichier précédent)

```
java -Xms4g -Xmx4g -cp $JOSHUA/bin/ \
   -Djava.library.path=$JOSHUA/lib \
   -Dfile.encoding=utf8 \
   joshua.zmert.ZMERT \
  -maxMem 500 /raid2/yannick/iwslt09/mert_syst3j/mert.config \
   > /raid2/yannick/iwslt09/mert_syst3j/log&
```

La pratique avec Joshua : décodage du test

- On re-extrait les règles de grammaires spécifiques au corpus de texte
- On utilise les poids optimisés sur le dev

La pratique avec Joshua : quelques résultats

- Résultats sur dev7 de IWSLT 09 (dev7 retiré des données d'optimisation des poids) en BLEU :
  - Moses : 53,41
  - Joshua : 54,00
  - Moses avec CSLM 54,75
  - SMT + Joshua : 55,54
  - SMTcslm + Joshua : 55,89