Extraktion von Preordering-Regeln für Maschinelle Übersetzung

Gruppe4 -Otedama-

Julian Hitschler, Benjamin Körner, Mayumi Ohta

Computerlinguistik
Universität Heidelberg
Softwareprojekt SS13

Übersicht

1. Problemstellung
2. Review
3. Ergebnisse
4. Reflektion

Übersicht

Problemstellung

Problem

- IBM Modelle bestrafen Reordering im Zielsatz
- Verursacht falsche Übersetzungen bei Sprachpaaren mit divergierender Syntax
  - head-initial(Englisch) vs. head-final(Japanisch)

Ansatz

- Reordering des Ausgangssatz als Preprocessing Step
- Heuristik: Minimierung des Crossing Score
  - Referenzpaper: Genzel, 2010[1]

Korpus

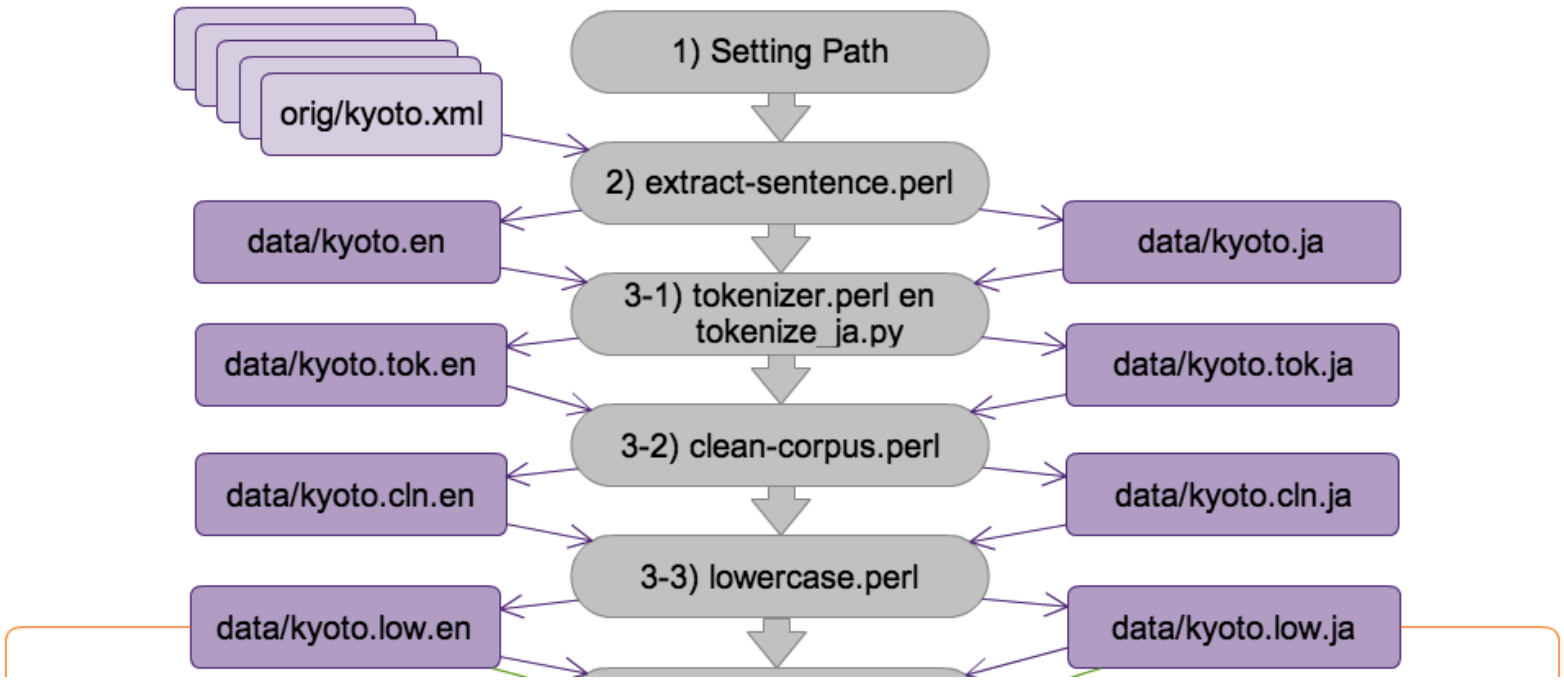
Kyoto Wiki English-Japanese Parallel Corpus

- stammt von Wikipedia Artikeln über die Stadt Kyoto
- enthält knapp halbe Million Satzpaare

set	articles	sentences	en words	ja words
Dev	15	1166	24309	24707
Test	15	1160	26734	26279
Train	14126	440286	11541358	10821659
Train (clean)	14126	343617	6365202	6065075
Tune	15	1235	30822	31409

Workflow

1) - 3) Preprocess



Workflow

kyoto.xml -> kyoto.en & kyoto.ja

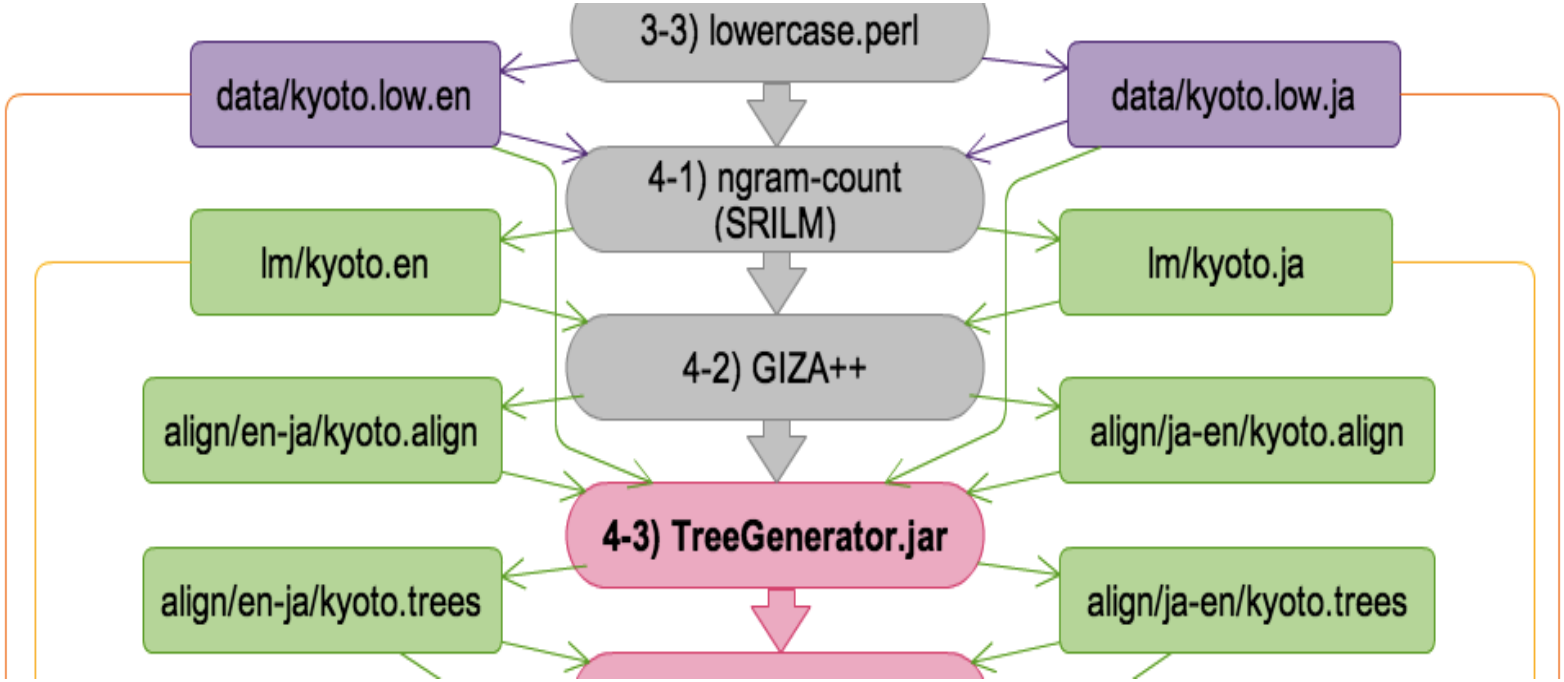
```
▼<sen id="2">
    <j>日本の水墨画を一変させた。</j>
    <e type="trans" ver="1">He changed Japanese ink painting.</e>
    <cmt/>
    <e type="trans" ver="2">He changed Japanese ink painting.</e>
    <cmt>修正なし</cmt>
    <e type="check" ver="1">He revolutionized the Japanese ink painting.</e>
    <cmt>フィードバックに基づき翻訳を修正しました。</cmt>
  </sen>
```

he revolutionized the japanese ink painting .
he was given the posthumous name " toyo " or " sesshu (拙宗) . "
born in bicchu
sshokoku-ji ter
later he accom
chinese ink pa
his works were
but also portra
his bold compo
style .6 of his e
he is considere

日本の水墨画を一変させた。
諱は「等楊（とうよう）」、もしくは「拙宗（せっしゅう）」と号した。
備中国に生まれ、京都・相国寺に入ってから周防国に移る。
その後遣明使に随行して中国（明）に渡って中国の水墨画を学んだ。
作品は数多く、中国風の山水画だけでなく人物画や花鳥画もよくした。
大胆な構図と力強い筆線は非常に個性的な画風を作り出している。
現存する作品のうち 6 点が国宝に指定されており、日本の画家のなかでも別格の評価を受け
このため、花鳥図屏風などに「伝雪舟筆」される作品は大変多い。
真筆であるか専門家の間でも意見の分かれるものも多々ある。
弟子に、秋月、宗淵、等春らがいる。
1420 年備中国赤浜（現在の岡山県総社市）に生まれる。

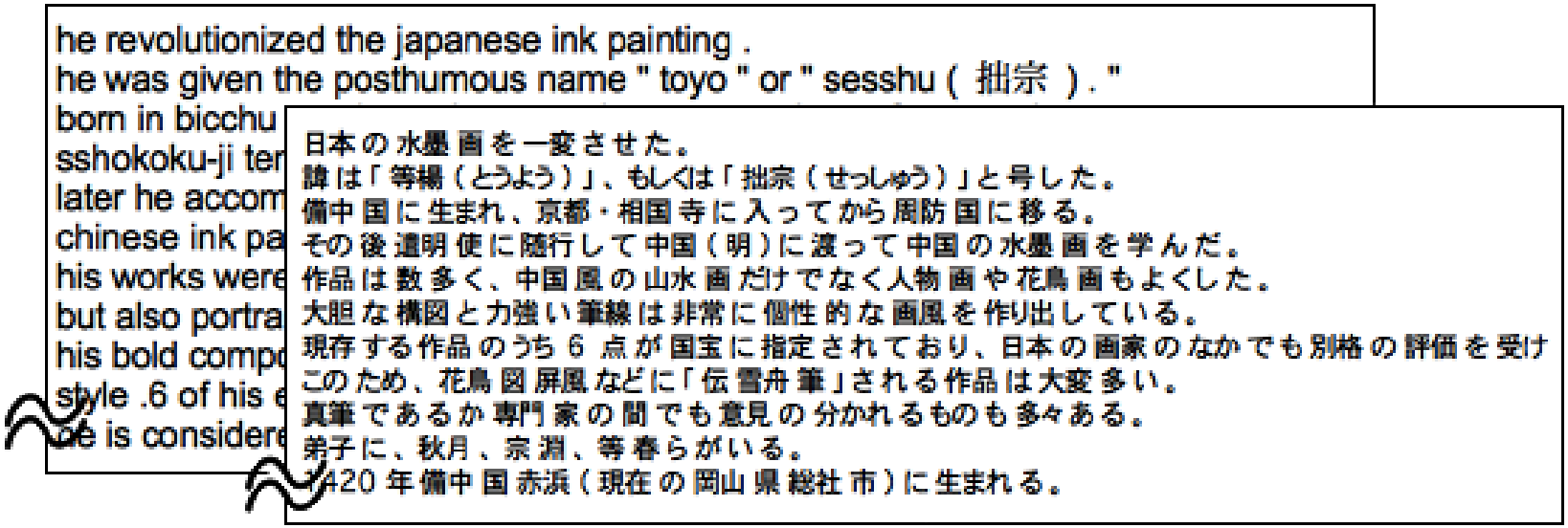
Workflow

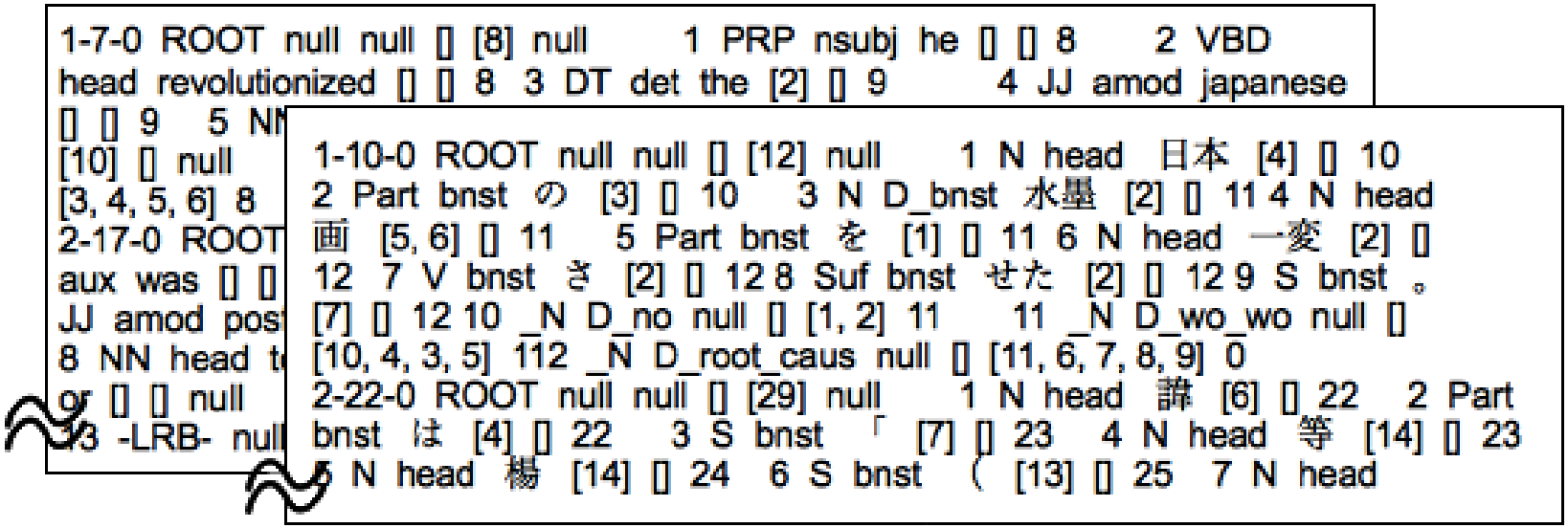
4) generate trees



Workflow

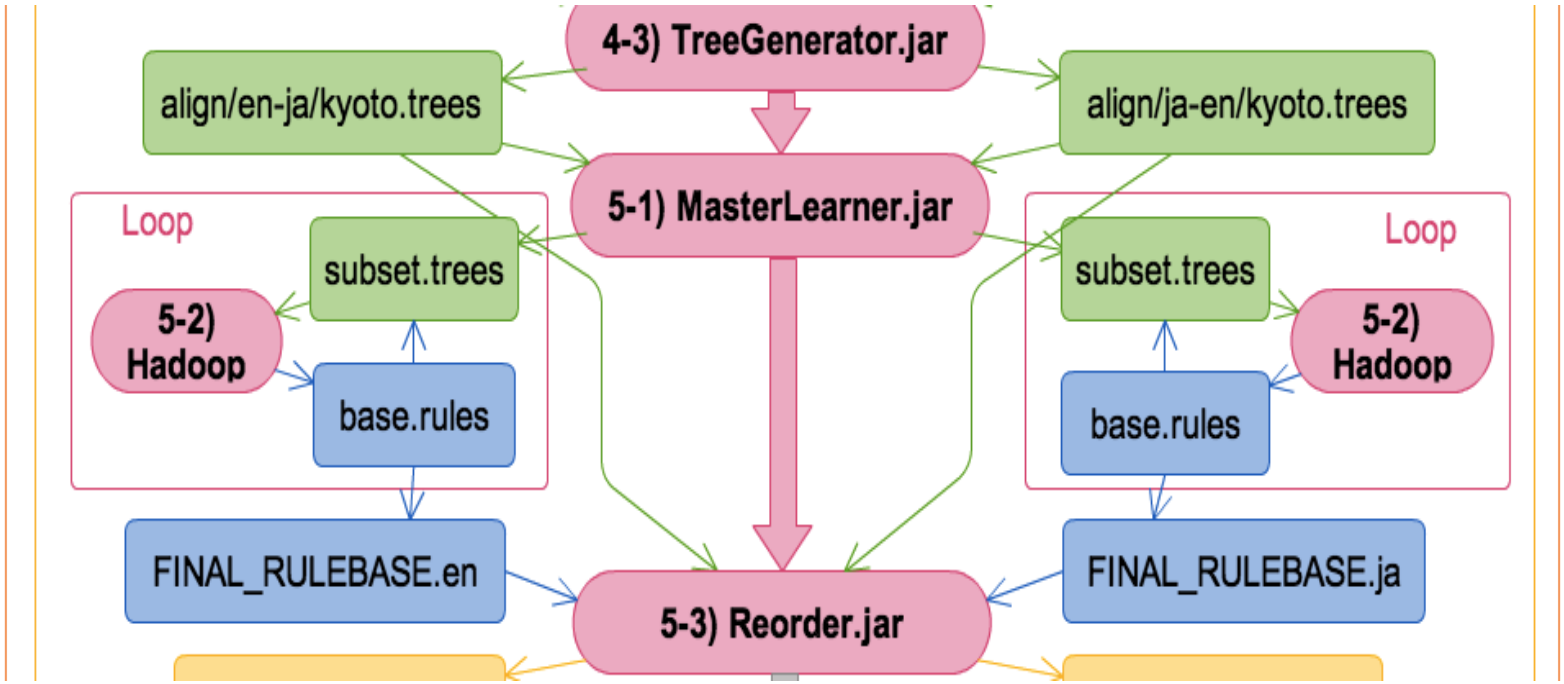
kyoto.en & kyoto.ja -> kyoto.en.trees & kyoto.ja.trees





Workflow

5) extract rules



Workflow

kyoto.trees -> FINALE_RULEBASE

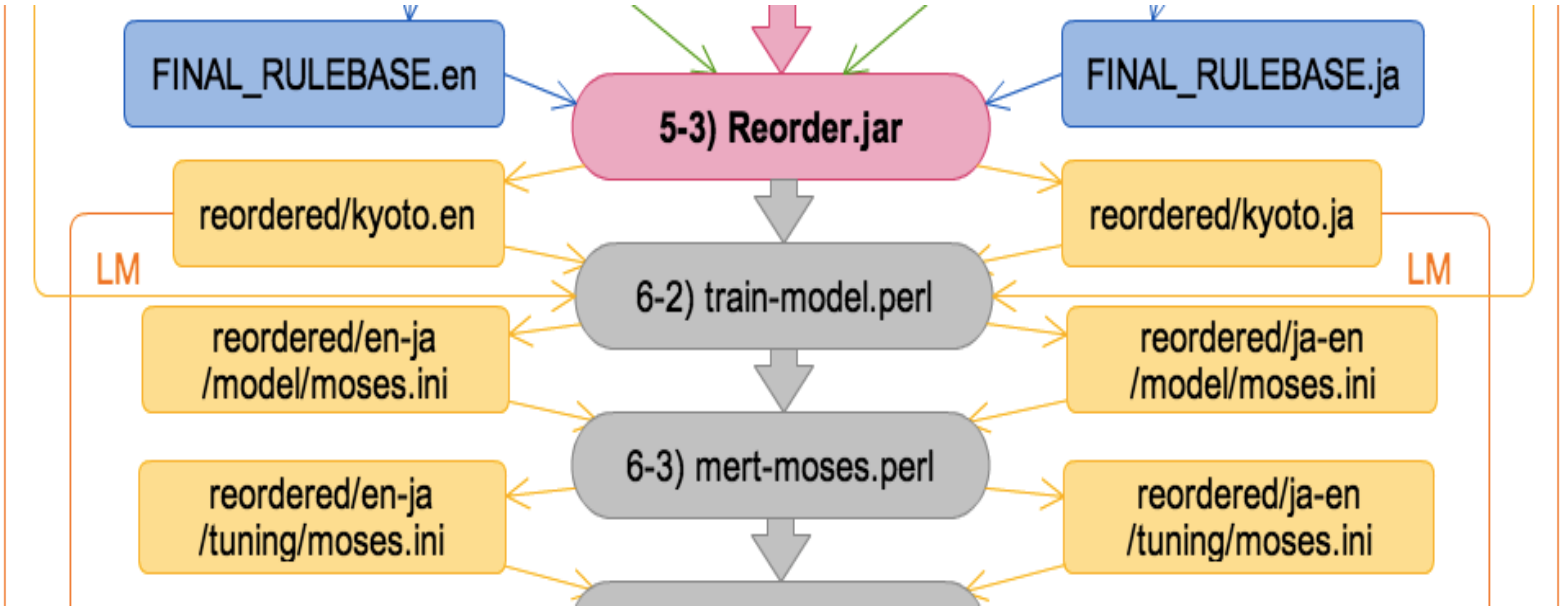
```
1-7-0 ROOT null null [] [8] null 1 PRP nsubj he [] [] 8 2 VBD
head revolutionized [] [] 8 3 DT det the [2] [] 9 4 JJ amod japanese
[] [] 9 5 NN
[10] [] null
[3, 4, 5, 6] 8
2-17-0 ROOT
aux was [] []
JJ amod pos
8 NN head t
or [] [] null
3 -LRB- null
```

```
1-10-0 ROOT null null [] [12] null 1 N head 日本 [4] [] 10
2 Part bnst の [3] [] 10 3 N D_bnst 水墨 [2] [] 11 4 N head
画 [5, 6] [] 11 5 Part bnst を [1] [] 11 6 N head 一変 [2] []
12 7 V bnst さ [2] [] 12 8 Suf bnst せた [2] [] 12 9 S bnst 。
[7] [] 12 10 _N D_no null [] [1, 2] 11 11 _N D_wo_wo null []
[10, 4, 3, 5] 112 _N D_root_caus null [] [11, 6, 7, 8, 9] 0
2-22-0 ROOT null null [] [29] null 1 N head 誹 [6] [] 22 2 Part
bnst は [4] [] 22 3 S bnst 「 [7] [] 23 4 N head 等 [14] [] 23
5 N head 楊 [14] [] 24 6 S bnst （ [13] [] 25 7 N head
```

```
1#{1L:auxpass;1T:VBZ;2L:head;2T:VBN;3L:xcomp;3T:_VB
;nL:root;nT:_VBN;pL:null;pT:ROOT}{1,2,3:3,1,2}{CROSSING:-1843.0}
2#{2L:auxpass;2T:VBD;3L:head;3T:VBN;4L:agent;4T:_NN
;nL:root;nT:_VBN;pL:null;pT:ROOT}{2,3,4:4,2,3}{CROSSING:-1221.0}
3#{0L:det;0T:DT;1L:head;1T:NNS;2L:prep_of;2T:NN
;nL:dobj;nT:_NNS;pL:prepc_by;pT:_VBG}{0,1,2:2,0,1}{CROSSING:-686.0}
4#{0L:nsubj;0T:NNP;1L:head;1T:VBD;2L:xcomp;2T:_NN
;nL:conj_and;nT:_VBD;pL:ccomp;pT:_VBD}{0,1,2:0,2,1}{CROSSING:-644.0}
```

Workflow

6) train and tune moses



Workflow

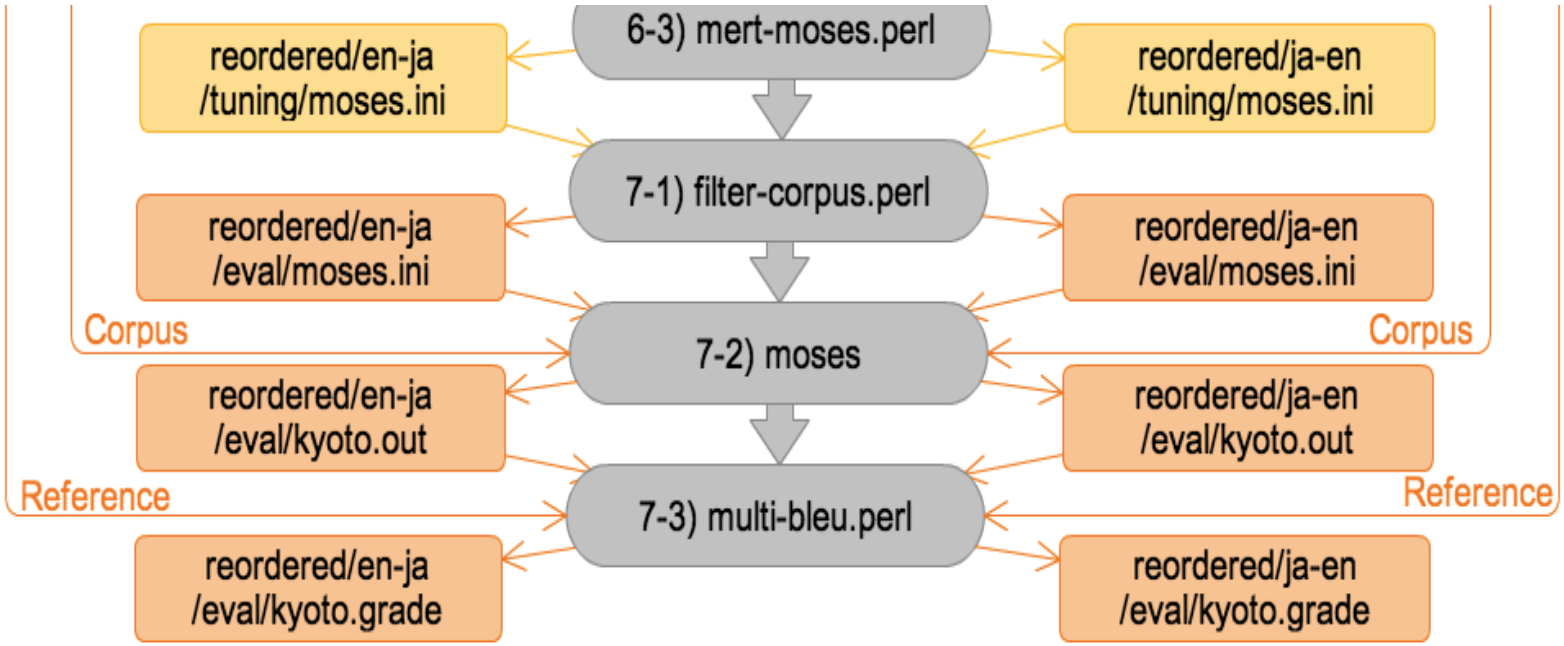
FINALE_RULEBASE -> moses.ini

```
1#{1L:auxpass;1T:VBZ;2L:head;2T:VBN;3L:xcomp;3T:_VB
;nL:root;nT:_VBN;pL:null;pT:ROOT}{1,2,3:3,1,2}{CROSSING:-1843.0}
2#{2L:auxpass;2T:VBD;3L:head;3T:VBN;4L:agent;4T:_NN
;nL:root;nT:_VBN;pL:null;pT:ROOT}{2,3,4:4,2,3}{CROSSING:-1221.0}
3#{0L:det;0T:DT;1L:head;1T:NNS;2L:prep_of;2T:NN
;nL:dobj;nT:_NNS;pL:prepc_by;pT:_VBG}{0,1,2:2,0,1}{CROSSING:-686.0}
4#{0L:nsubj;0T:NNP;1L:head;1T:VBD;2L:xcomp;2T:_NN
;nL:conj_and;nT:_VBD;pL:ccomp;pT:_VBD}{0,1,2:0,2,1}{CROSSING:-644.0}
```

```
# MERT optimized configuration
# decoder /usr/local/bin/moses
# BLEU 0.150582 on dev /kftt-moses-1.4/data/low/kyoto-tune.ja
# We were before running iteration 14
# finished Mo 10 Jun 2013 16:06:30 CEST
###############################
### MOSES CONFIG FILE ###
###############################
```

Workflow

7) test and evaluate



Workflow

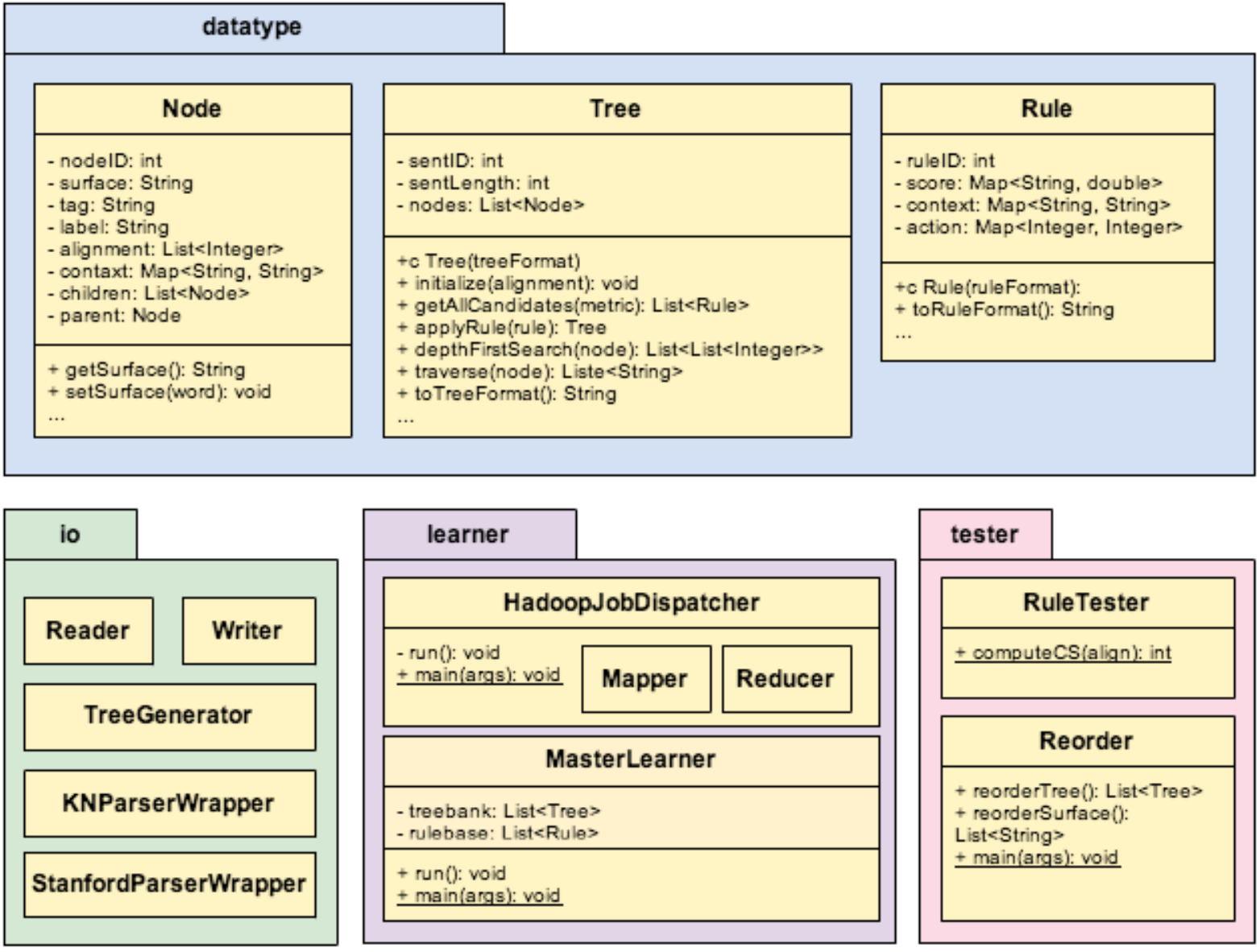
moses.ini -> test.out & test.grade

```
# MERT optimized configuration
# decoder /usr/local/bin/moses
# BLEU 0.150582 on dev /kftt-moses-1.4/data/low/kyoto-tune.ja
# We were before running iteration 14
# finished Mo 10 Jun 2013 16:06:30 CEST
################################
### MOSES CONFIG FILE ###
################################
```

```
infoboxbudd hist
dogen ( どうげん ) was a zen monk in the early kamakura period .
the four
in his la
in the h
his post
it is gen
in japan
the first
```

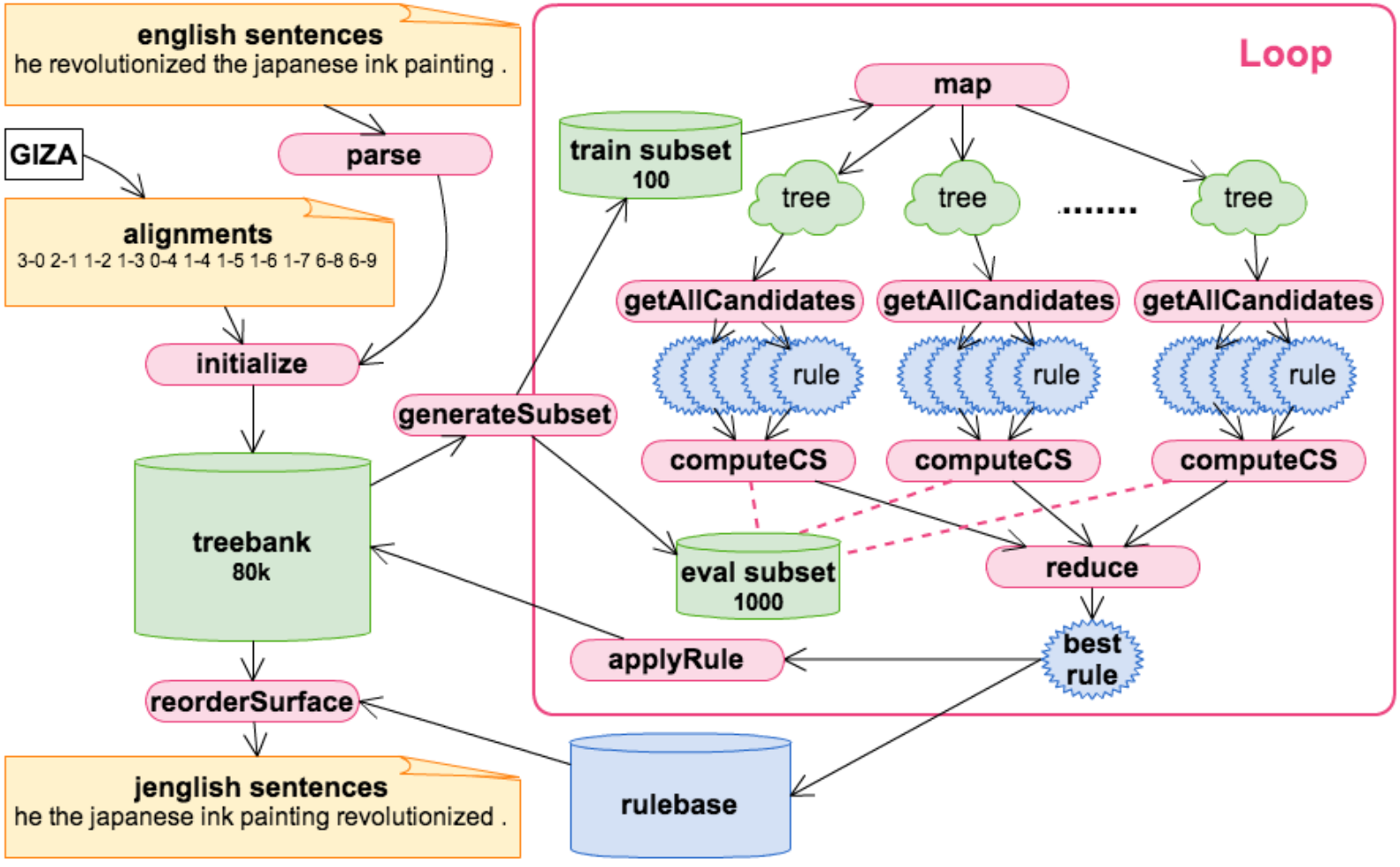
```
BLEU = 10.23, 52.0/18.4/8.2/4.0 (BP=0.768, ratio=0.791, hyp_len=22713,
ref_len=28698)
```

Systemarchitektur

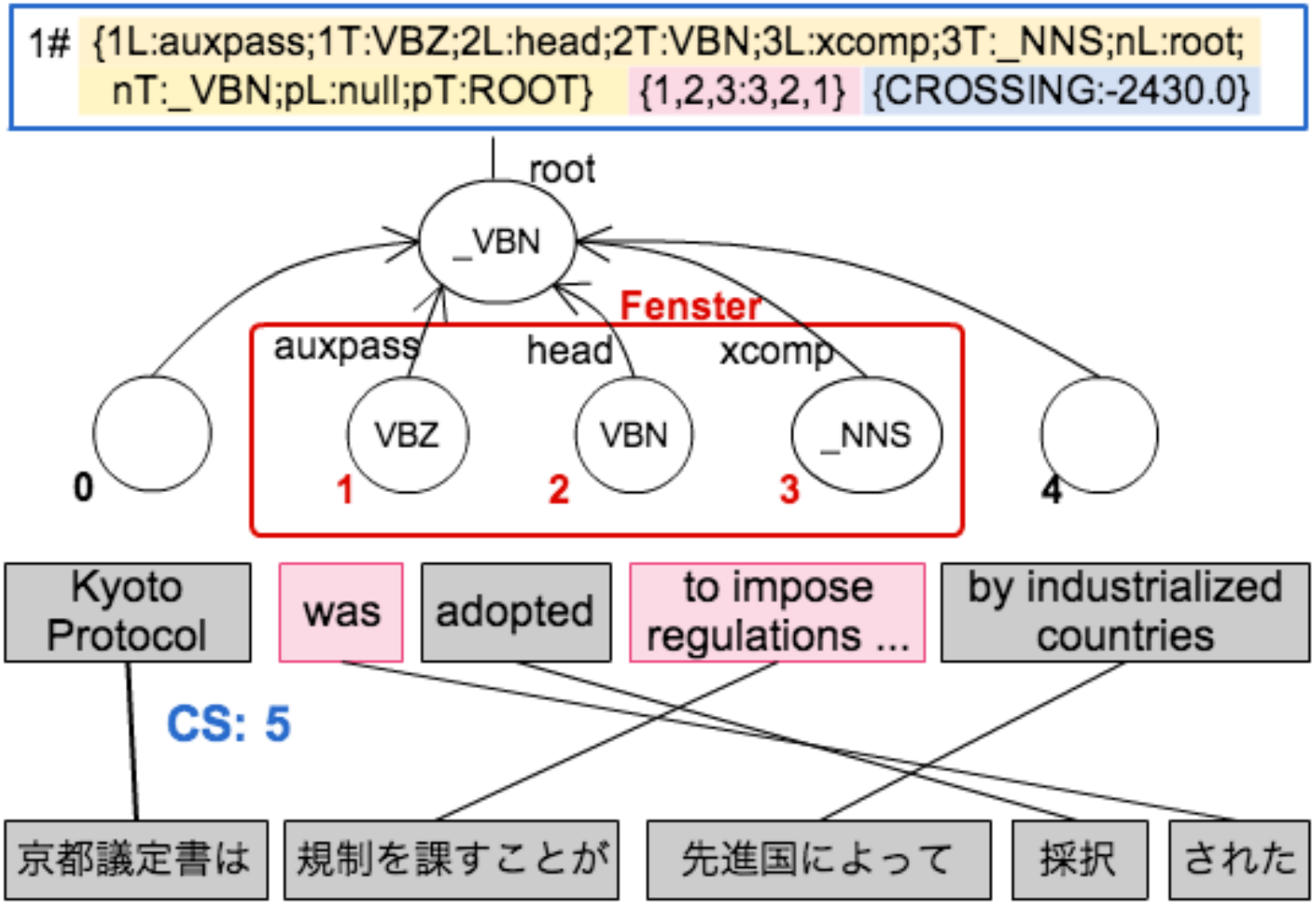


Übersicht

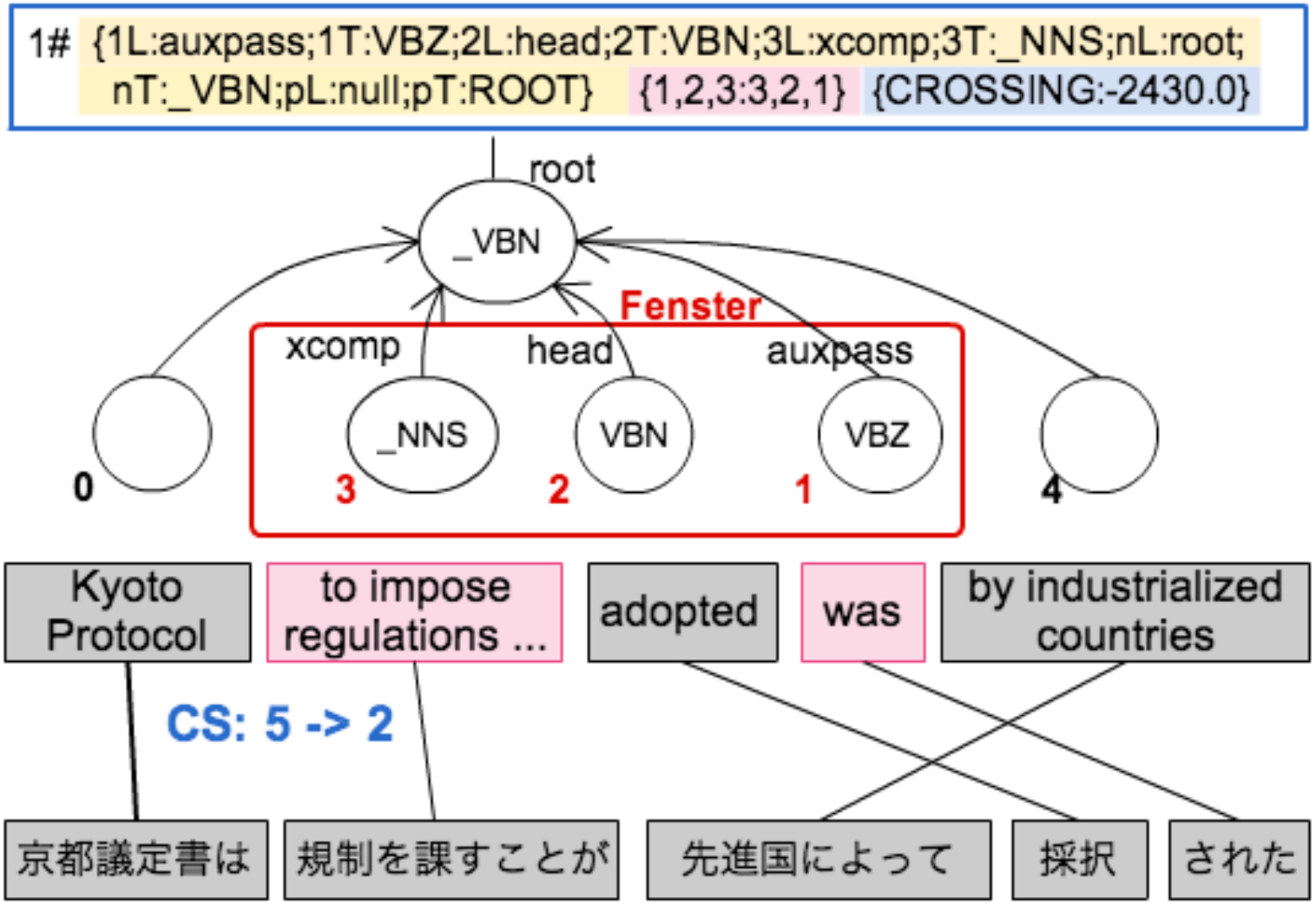
Algorithmus



Beispielregel



Beispielregel



Korpus

Was haben wir gleich und anders als Genzel gemacht?

Übersetzungsrichtung

- Englisch → Japanisch als auch Japanisch → Englisch
  Genzel: nur Englisch → Japanisch (und andere Sprachpaare)

Umfang

- 6.3M Wörter als Trainingsset
  Genzel: 28M - 260M Wörter pro Sprache als Trainingsset

Regeln

Was haben wir gleich und anders als Genzel gemacht?

Generalisierung

- nur eine Regel pro Iteration
  Genzel: top k Regeln mit der Elimination der Überlappung

  → enorme Vereinfachung des Algorithmus
  → größere Robustheit des gelernten Regelsets

Metrik

- Evaluation auf separatem Random Subset
- kumulative Differenz zwischen vor und nach Umordnung
- *estimated BLEU Score* nicht verfolgt

Matching

- weniger strenge Kriterien (nur bis 4 Features)
  Genzel: kleine Featuretabelle aber strengeres Matching

Bäume

Was haben wir gleich und anders als Genzel gemacht?

Features

- Stanford Dependency Types und POS für das Englische
- ausgewählte Kasuslabels des KNParsers für das Japanische

Alignments

- nur bis IBM Model 1

  → weniger Bias zu Monotonizität
- unwahrscheinlichen Alignments nicht gelöscht

  → sprachspezifisch, Art des Kontext-Matchings?

Umordnung

- mit Tiefensuche
- nur bis 3 Kindknoten auf einmal (Fensterverschiebung)

Übersicht

Ergebnisse: en → ja

n-gram order	distortion limit	rule extraction trainset size	rule size	tune	en→ ja BLEU	en→ ja baseline
3	0	50k	20	-	14.74	10.23
3	5	50k	20	-	14.74	12.40
3	-1	50k	20	-	15.09	12.52
3	-1	80k	20	-	14.44	12.52
3	5	80k	20	-	14.11	12.40
3	5	80k	30	-	14.29	12.40
3	5	80k	50	-	13.59	12.40
3	5	80k	100	-	13.78	12.40
3	5	80k	200	-	14.30	12.40

Ergebnisse: ja → en

n-gram order	distortion limit	feature size	train size	rule size	tune	en→ ja BLEU	en→ ja baseline
3	0	4	80k	20	+	14.11	15.88
3	5	4	80k	20	+	15.72	17.21
3	-1	4	80k	20	+	16.81	18.92
3	0	4	80k	50	+	14.15	15.88
3	0	4	80k	100	+	14.14	15.88
3	0	4	80k	200	+	13.61	15.88
3	0	4	80k	500	+	13.74	15.88
3	0	8	80k	20	+	11.84	15.88
3	5	8	80k	20	+	13.66	17.21
3	-1	8	80k	20	+	14.95	18.92

Evaluation

Baseline (ohne Reordering)

Source: 道 元 （ どう げん ） は 、 鎌倉 時代 初期 の 禅 僧 。

Output: dogen (or) in the early kamakura period) was a zen monk .

Jenglish (mit Reordering)

Source: ） は 道 元 （ どう げん 、 僧 禅 。 の 鎌倉 時代 初期

Output: the dogen (genho was a zen monk in the early kamakura period.

Goldstandard

dogen was a zen monk in the early kamakura period .

Fazit und Ausblick

- Algorithmus vereinfacht
- Algorithmus funktioniert auch auf kleinem Datenset
- Ausprobieren der Regelsets auf parellelen Korpora “handelsüblicher” Größe
- Ergebnisse JP - EN stehen noch aus
- Vergleich Estimated BLEU / Crossing Score
- Mehr Daten für die Learning Curve
- Ausprobieren von liberalerem / strengerem Kontext-Matching (3, 5 Features, etc.)

Übersicht

Lessons Learned

Entwicklungsumgebung

- SVN/Google Code via Eclipse hat sich als Versionskontrollsystem bewährt

Kommunikation

- hat gut über Email funtioniert
- Foren / Wikis bei Projekten dieser Größe nicht nötig

Projektmanagement

- Arbeitseinteilung hat gut funktioniert und war angemessen
- eventuell mehr Code Reviews / festere Deadlines

Ablauf

<table>
<tr><th></th><th>Plan</th><th>Ablauf</th></tr>
<tr><td>Mai</td><td>Recherche, Planung</td><td>Recherche, Planung</td></tr>
<tr><td>4. Juni</td><td>Datentypen, Fileformate</td><td>Datentypen, Fileformate</td></tr>
<tr><td>11. Juni</td><td rowspan="3">Vollständiges System
- EN → JP mit CS
- EN → JP mit BLEU
- JP → EN</td><td rowspan="4">Vollständiges System
- EN → JP mit CS

- auf Pseudo-Hadoop
- auf echtem Hadoop</td></tr>
<tr><td>18. Juni</td></tr>
<tr><td>25. Juni</td></tr>
<tr><td>2. Juli</td><td>Weitere Sprachen</td></tr>
<tr><td>9. Juli</td><td>Optimierung</td><td>Workflow-Scripting</td></tr>
<tr><td>16. Juli</td><td>Experimente, Test</td><td>JP → EN, Experimente</td></tr>
<tr><td>23. Juli</td><td>Dokumentation</td><td>Dokumentation, Test</td></tr>
<tr><td>30. Juli</td><td>Abgabe</td><td>Abgabe??</td></tr>
</table>

Special Thanks!

Graham Neubig (NAIST)
hat ein tolles Moses-Script veröffentlicht!

Gruppe3
hat netterweise ihr kompiliertes Moses auch für uns zur Verfügung gestellt!

Hiko Schamoni
hat uns vor allem bei Hadoop-Problemen geholfen!

Laura Jehl
hat uns so umfassend betreut!

Literatur

D. Genzel.
Automatically Learning Source-side Reordering Rules for Large Scale Machine Translation.
Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics ACL 2010, (August):376-384, 2010.

M. Collins, et. al.
Clause restructuring for statistical machine translation.
Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics ACL05, (June):531-540, 2005.

Data & Tools

Kyoto Free Translation Task version 1.4

http://www.phontron.com/kftt/index.html

Moses version 1.0

http://www.statmt.org/moses/

SRILM version 1.7.0

http://www.speech.sri.com/projects/srilm/

GIZA++ version 1.0.7

https://code.google.com/p/giza-pp/

Stanford Parser version 2.0.5

http://nlp.stanford.edu/software/lex-parser.shtml

KN Parser version 4.1

http://nlp.ist.i.kyoto-u.ac.jp/EN/index.php?KNP