MaxForce: Max-Violation Perceptron and Forced Decoding for Scalable MT Training

Bush held talks with Sharon

Heng Yu  Liang Huang  Haitao Mi  Kai Zhao

Chinese Acad. of Sciences  CUNY  IBM T. J. Watson  CUNY
MaxForce: Max-Violation Perceptron and Forced Decoding for Scalable MT Training

Heng Yu
Chinese Acad. of Sciences

Liang Huang
CUNY

Haitao Mi
IBM T. J. Watson

Kai Zhao
CUNY

Scalable Training for MT Finally Made Successful

Bush held talks with Sharon

Heng Yu    Liang Huang    Haitao Mi    Kai Zhao

Chinese Acad. of Sciences    CUNY    IBM T. J. Watson    CUNY
Discriminative Training for SMT

- Discriminative training is dominant in parsing / tagging
- Can use arbitrary, overlapping, lexicalized features
- But not very successful yet in machine translation
- Most efforts on MT training tune feature weights on the small dev set (~1k sents) not the training set!
- As a result can only use ~10 dense features (MERT)
- Or ~10k rather impoverished features (MIRA/PRO)
- Liang et al (2006) train on the training set but failed

training set (>100k sentences)  dev set (~1k sents)  test set (~1k sents)
Timeline for MT Training

- **Training set**: >100k sentences
- **MERT**: (Och ’02) (dense features)
- **Dev set**: ~1k sents
- **Test set**: ~1k sents
Timeline for MT Training

Standard Perceptron (a noble failure)  
(Liang et al. 2006)

MERT  
(Och ’02)  
(dense features)

training set (>100k sentences)  
dev set  
(~1k sents)  
test set  
(~1k sents)
Timeline for MT Training

**Standard Perceptron** (a noble failure)
(Liang et al 2006)

**MERT**
(Och ’02) (dense features)

**MIRA**
(Watanabe+ ’07)
(Chiang+ ’08-’12) (pseudo sparse features)

training set (>100k sentences)

dev set (~1k sents)

test set (~1k sents)
Timeline for MT Training

- **Training set (>100k sentences)**
- **Dev set (~1k sents)**
- **Test set (~1k sents)**

**Standard Perceptron** (a noble failure)  
(Liang et al 2006)

**MERT**  
(Och '02)

**MIRA**  
(Watanabe+ '07)  
(Chiang+ '08-'12)

**PRO**  
(Hopkins+May '11)

**Regression**  
(Bazrafshan+ '12)

(dense features)

(pseudo sparse features)
Timeline for MT Training

Standard Perceptron (a noble failure) (Liang et al 2006)

MERT (Och '02)

MIRA (Watanabe+ '07)
(Chiang+ '08-'12)

PRO (Hopkins+May '11)
Regression (Bazrafshan+ '12)

HOLS (Flanigan+ '13)
(sparse features as one dense feature)

training set (>100k sentences)

dev set (~1k sents)

test set (~1k sents)
Timeline for MT Training

**Standard Perceptron** (a noble failure) (Liang et al 2006)

**MERT** (Och ’02)
(dense features)

**MIRA** (Watanabe+ ’07)
(Chiang+ ’08’12)
(pseudo sparse features)

**PRO** (Hopkins+May ’11)
Regression
(Bazrafshan+ ’12)

**HOLS** (Flanigan+ ’13)
(sparse features as one dense feature)

**our work (2013): violation-fixing perceptron with truly sparse features**

**Training set (>100k sentences)**

**Dev set (~1k sents)**

**Test set (~1k sents)**
Timeline for MT Training

Standard Perceptron (a noble failure)  
(Liang et al 2006)

MERT  
(Och ’02)  
(dense features)

MIRA  
(Watanabe+ ’07)  
(Chiang+ ’08-’12)  
(pseudo sparse features)

PRO  
(Hopkins+May ’11)  
Regression  
(Bazrafshan+ ’12)  
(sparse features as one dense feature)

HOLS  
(Flanigan+ ’13)

our work (2013): violation-fixing perceptron with truly sparse features

training set (>100k sentences)

dev set  
(~1k sents)

test set  
(~1k sents)
Why previous work fails

- their learning methods are based on exact search
  - MT has huge search spaces => severe search errors
    - learning algorithms should fix search errors
    - full updates (perceptron/MIRA/PRO) can’t fix search errors
  - MT involves latent variables (derivations not annotated)
    - perceptron/MIRA was not designed for latent variables
- we need better variants for perceptron
Why our approach works

- use a variant of perceptron tailored for inexact search
- fix search errors in the middle of the search
- “partial updates” instead of “full updates”
- use forced decoding lattice as the target to update to
- use parallelized minibatch to speed up learning
- result: scaled to a large portion of the training data
  - 20M sparse features => +2.0 BLEU over MERT/PRO
MT as Structured Classification

- with latent variables (hidden derivations)

\[
\begin{array}{c}
\text{那人 咬 了 狗} \\
\text{the man bit the dog}
\end{array}
\]
MT as Structured Classification

- with latent variables (hidden derivations)

All gold derivations
MT as Structured Classification

- with latent variables (hidden derivations)

all gold derivations
MT as Structured Classification

• with latent variables (hidden derivations)

all gold derivations

[Diagram showing the man bit the dog and the dog bit the man with arrows indicating derivations]
MT as Structured Classification

- with latent variables (hidden derivations)

![Diagram showing multiple translations with arrows pointing to gold derivations and a wrong translation.]

- all gold derivations

- best derivation

- wrong translation
MT as Structured Classification

- with latent variables (hidden derivations)

all gold derivations

wrong translation
MT as Structured Classification

- with latent variables (hidden derivations)

all gold derivations

update: penalize best derivation and reward best gold derivation
Outline

• Motivations

• Phrase-based Translation and Forced Decoding

• Violation-Fixing Perceptron for SMT
  • Update Strategies: Early Update and Max-Violation
  • Feature Design

• Experiments
Phrase-based translation

布什 与 沙龙 举行 了 会谈

Bushi yu Shalong juxing le huitan

held talks with Sharon

Bush with Sharon held meetings

Bushi yu Shalong juxing le huitan
Bushi held talks with Sharon.
Phrase-based translation

布什 与 沙龙 举行 了 会谈

Bushi yu Shalong juxing le huitan

held talks with Sharon

Bush with Sharon held meetings

Bushi yu Shalong juxing le huitan

held talks with Sharon
Bushi held talks with Sharon.
Bushi held talks with Sharon.
Bush held talks with Sharon.
Bushi held talks with Sharon.
Language Model and Beam Search

- split each -LM state into many +LM states
Language Model and Beam Search

- split each -LM state into many +LM states
Language Model and Beam Search

- split each -LM state into many +LM states

- Bush

  - ... meeting
  - ... talks
  - ... talk
Language Model and Beam Search

- split each -LM state into many +LM states

- Bush
  - ... meeting
  - ... talks
  - ... talk

- ... Shalong
- ... Sharon
Language Model and Beam Search

- split each -LM state into many +LM states

Bush

... meeting

... talks

... talk

... Sharon

... Shalong
Forced Decoding

- both as data selection (more literal) and oracle derivations

Bushi yu Shalong juxing le huitan  →  Bush held talks with Sharon

gold derivation lattice  held talks  with Sharon

Bush  held  talks  with Sharon
Forced Decoding

- both as data selection (more literal) and oracle derivations

*Bushi yu Shalong juxing le huitan*  ➔  *Bush held talks with Sharon*

gold derivation lattice

0 1 2 3 4 5 6
Forced Decoding

- both as data selection (more literal) and oracle derivations

Bushi yu Shalong juxing le huitan  \(\rightarrow\)  Bush held talks with Sharon

- gold derivation lattice
- held talks
- with Sharon
Forced Decoding

- both as data selection (more literal) and oracle derivations

**Bushi yu Shalong juxing le huitan** → **Bush held talks with Sharon**

![Diagram of forced decoding process withBush, Sharon, meeting, talks, and talk nodes linked by arrows and dots representing derivations.](image)
Forced Decoding

- both as data selection (more literal) and oracle derivations

*Bushi yu Shalong juxing le huitan* → *Bush held talks with Sharon*

Gold derivation lattice

- held talks
- with Sharon

Bush: 

- **Bush**
- **held**
- **talks**
- **with** Sharon

1. Bush
2. held
3. talks
4. with Sharon
5. Sharon
6. Shalong
Forced Decoding

- both as data selection (more literal) and oracle derivations

*Bushi yu Shalong juxing le huitan*  
*Bush held talks with Sharon*

---

Bushi yu Shalong juxing le huitan  
Bush held talks with Sharon

---

gold derivation lattice  
held talks  
with Sharon

---

Bush  
held  
talks  
with Sharon

---

0 1 2 3 4 5 6
Forced Decoding

- both as data selection (more literal) and oracle derivations

Bush held talks with Sharon

Bushi yu Shalong juxing le huitan

one gold derivation

gold derivation lattice

held talks

with Sharon
distortion limit causes unreachability (hiero would be better)

but we can still use reachable prefix-pairs of unreachable pairs
Unreachable Sentences and Prefix

- distortion limit causes unreachability (hiero would be better)
- but we can still use reachable prefix-pairs of unreachable pairs
how many sentences pairs pass forced decoding?

the ratio drops dramatically as sentences get longer

prefixes boost coverage
Sentence/Word Reachability Ratio

- how many sentences pairs pass forced decoding?
- the ratio drops dramatically as sentences get longer
- prefixes boost coverage

<table>
<thead>
<tr>
<th></th>
<th>small</th>
<th></th>
<th>large</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent.</td>
<td>words</td>
<td>sent.</td>
<td>words</td>
</tr>
<tr>
<td>full</td>
<td>21.4%</td>
<td>8.8%</td>
<td>32.1%</td>
<td>12.7%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
<td>24.6%</td>
<td>67.3%</td>
<td>32.8%</td>
</tr>
</tbody>
</table>
Sentence/Word Reachability Ratio

- how many sentences pairs pass forced decoding?
- the ratio drops dramatically as sentences get longer
- prefixes boost coverage

<table>
<thead>
<tr>
<th></th>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent.</td>
<td>words</td>
</tr>
<tr>
<td>full</td>
<td>21.4%</td>
<td>8.8%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
<td>24.6%</td>
</tr>
</tbody>
</table>

![Graph showing reachability ratio vs. sentence length]

- Distortion-unlimit
- Distortion-limit 6
- Distortion-limit 4
- Distortion-limit 2
- Distortion-limit 0

![Graph showing reachability ratio vs. sentence length, inset showing different distortion limits]
Number of Gold Derivations

- exponential in sentence length (on fully reachables)
- these are the “latent variables” in learning

![Graph showing the average number of derivations vs sentence length for different distances (dist-6, dist-4, dist-2, dist-0). The graph illustrates an exponential increase in derivations with sentence length.]
Outline

• Background: Phrase-based Translation  (Koehn, 2004)

• Forced Decoding

• Violation-Fixing Perceptron for MT Training
  • Update strategy
  • Feature design

• Experiments
Structured Perceptron (Collins 02)

binary classification

\[
x \quad y=+1 \\
x \quad y=-1
\]

update weights if \( y \neq z \)

exact inference

15
Structured Perceptron (Collins 02)

Binary classification

Structured classification

\[ y = 1 \]
\[ y = -1 \]
Structured Perceptron (Collins 02)

**Binary classification**

\[ y = +1 \quad y = -1 \]

**Structured classification**

那 人 咬 了 狗
the man bit the dog

**Diagram**

- **Exact inference**
- **Update weights if** \( y \neq z \)
challenges in applying perceptron for MT

- the inference (decoding) is vastly inexact (beam search)
- we know standard perceptron doesn’t work for MT
- intuition: the learner should fix the search error first
Structured Perceptron (Collins 02)

- challenges in applying perceptron for MT
  - the inference (decoding) is vastly *inexact* (beam search)
  - we know standard perceptron doesn’t work for MT
- intuition: the learner should fix the search error first
Bush held talks with Sharon

gold derivation lattice

real decoding beam search
Search Error: Gold Derivations Pruned

gold derivation lattice

Bush held talks with Sharon

held talks

real decoding beam search
Search Error: Gold Derivations Pruned

gold derivation lattice

Bush

held talks

with Sharon

real decoding beam search
Search Error: Gold Derivations Pruned

gold derivation lattice

real decoding beam search

Bush held talks with Sharon

held talks with Sharon

0 1 2 3 4 5 6
Search Error: Gold Derivations Pruned

gold derivation lattice

Bush

held talks

held

talks

with Sharon

real decoding beam search
Search Error: Gold Derivations Pruned

gold derivation lattice

held talks

Bush

held

talks

with Sharon

real decoding beam search
Search Error: Gold Derivations Pruned

gold derivation lattice

held talks

with Sharon

Bush

held

talks

with

Sharon

real decoding beam search

should fix search errors here!
Fixing Search Error 1: Early Update

Model

standard update (no guarantee!)
Fixing Search Error 1: Early Update

• early update \((\text{Collins/Roark'04})\) when the correct falls off beam

• up to this point the incorrect prefix should score higher

• that’s a “violation” which we want to fix
Fixing Search Error 1: Early Update

- early update \((Collins/Roark'04)\) when the correct falls off beam
- up to this point the incorrect prefix should score higher
- that’s a “violation” which we want to fix
Fixing Search Error 1: Early Update

- early update (Collins/Roark’04) when the correct falls off beam
  - up to this point the incorrect prefix should score higher
  - that’s a “violation” which we want to fix
Fixing Search Error 1: Early Update

- early update (Collins/Roark’04) when the correct falls off beam
- up to this point the incorrect prefix should score higher
- that’s a “violation” which we want to fix

![Diagram showing the correct and incorrect prefixes and the impact of early update]
Fixing Search Error 1: Early Update

- early update (Collins/Roark’04) when the correct falls off beam
- up to this point the incorrect prefix should score higher
- that’s a “violation” which we want to fix
- standard perceptron does not guarantee violation
- w/ pruning, the correct seq. might score higher at the end!
- called “invalid” update b/c it doesn’t fix the search error
- the gold-standard derivations are not annotated
- we treat any reference-producing derivation as good

gold derivation lattice

Bush held talks with Sharon

Model
the gold-standard derivations are **not** annotated

- we treat any reference-producing derivation as good

gold derivation lattice
Early Update w/ Latent Variable

- the gold-standard derivations are **not** annotated
- we treat any reference-producing derivation as good

gold derivation lattice

Bush held talks with Sharon

0 1 2 3 4 5 6

Model

correct
the gold-standard derivations are **not** annotated

we treat any reference-producing derivation as good gold derivation lattice

- Bush held talks with Sharon
- Model
the gold-standard derivations are not annotated

we treat any reference-producing derivation as good
gold derivation lattice

Model

all correct derivations fall off
Early Update w/ Latent Variable

- the gold-standard derivations are **not** annotated
- we treat any reference-producing derivation as good

gold derivation lattice

---

Model

---

violation guaranteed:
incorrect *prefix* scores
higher up to this point

all correct derivations fall off
Early Update w/ Latent Variable

- the gold-standard derivations are **not** annotated
- we treat any reference-producing derivation as good

Gold derivation lattice

- violation guaranteed: incorrect prefix scores higher up to this point
- all correct derivations fall off
- stop decoding
• early update works but learns slowly due to partial updates

• **max-violation**: use the prefix where violation is maximum

• “worst-mistake” in the search space

• we call these methods “violation-fixing perceptrons” (Huang et al 2012)
Early Update vs. Max-Violation

- Early Update
- Must be invalid
- Max-violation
- Standard Update

Best in the beam: $d_i^-$
Worst in the beam: $d_i^+$
Max-violation: $d_i^*$
Standard update is invalid:

Model $w$

0 1 2 3 4 5 6
Early Update vs. Max-Violation

- **Early Update** vs. **Max-Violation**
- **Model w**
- **Best in the beam**
- **Worst in the beam**
- Standard update is invalid

The diagram illustrates the comparison between early updates and max-violation updates in a model. The x-axis represents different time steps (0-6), and the y-axis represents model weight. The diagram shows how different updates affect the model's performance across these steps.
Early Update vs. Max-Violation

best in the beam

worst in the beam

model $w$

$\frac{d_{i}^{-}}{d_{i}^{+}}$

$\frac{d_{i}^{-*}}{d_{i}^{+*}}$

$\frac{d_{i}^{y}}{d_{i}^{x}}$

$\frac{d_{i}^{x}}{d_{i}^{y}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

standard update is invalid

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$

$\frac{d_{i}^{-}}{d_{i}^{x}}$

$\frac{d_{i}^{+}}{d_{i}^{x}}$
Early Update vs. Max-Violation

best in the beam
\(d_i^-\)
\(d_i^*\)
worst in the beam
\(d_i^+\)

\(d_{i|x}^+\)
\(d_{i|x}^-\)

\(d_{i|\text{std}}^y\)

standard update is invalid

model \(w\)

0 1 2 3 4 5 6
Early Update vs. Max-Violation

Model $w$

Best in the beam $d_i^-$

Max-violation $d_i^* - d_i^+$

Worst in the beam $d_i^+$

Standard update is invalid

Early-update
Early Update vs. Max-Violation

- Early-update
- Max-violation

(best in the beam)

\[ d_i^- \]

\[ d_i^* \]

worst in the beam

\[ d_i^+ \]

\[ d_i^+ \]

\[ d_{|x|}^y \]

\[ d_{|x|}^- \]

standard update is invalid

model \( w \)

local

std

- Early-update
Early Update vs. Max-Violation

best in the beam

worst in the beam

model w

early

max-violation

std

local

standard update is invalid

Early-update

0 1 2 3 4 5 6
Early Update vs. Max-Violation

- Early-update
- Max-violation

Model $w$

- Best in the beam
- Worst in the beam

$d_i^-$ early $d_i^-$

$d_i^+$ max-violation $d_i^+$

$d_{(x)}^y$ local

$d_{(x)}$ std

Standard update is invalid

Early-update
Early Update vs. Max-Violation

![Diagram showing the comparison between early update and max-violation methods in a beam search. The figure illustrates the model's progress through the beam, with early updates being considered at each step, and max-violation updates being applied when the current update is invalid.](image-url)
Early Update vs. Max-Violation

- Early-update
- Max-violation

- best in the beam
- worst in the beam

- standard update is invalid
Latent-Variable Perceptron

correct sequence

full (standard)

invalid update!

standard update is invalid

model $w$

best in the beam

worst in the beam

correct sequence
Roadmap of the techniques

structured perceptron (Collins, 2002)
Roadmap of the techniques

structured perceptron (Collins, 2002)

latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)
Roadmap of the techniques

- **structured perceptron** (Collins, 2002)
  - **latent-variable perceptron** (Zettlemoyer and Collins, 2005; Sun et al., 2009)
  - **perceptron w/ inexact search** (Collins & Roark, 2004; Huang et al. 2012)
Roadmap of the techniques

- structured perceptron (Collins, 2002)
- latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)
- perceptron w/ inexact search (Collins & Roark, 2004; Huang et al. 2012)
- latent-variable perceptron w/ inexact search (Yu et al. 2013)
Roadmap of the techniques

structured perceptron (Collins, 2002)

latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)

perceptron w/ inexact search (Collins & Roark, 2004; Huang et al. 2012)

latent-variable perceptron w/ inexact search (Yu et al. 2013)

hiero  syntactic parsing  semantic parsing  transliteration
Feature Design

- **Dense features:**
  - standard phrase-based features (Koehn, 2004)

- **Sparse Features:**
  - rule-identification features (unique id for each rule)
  - word-edges features
    - lexicalized local translation context within a rule
  - non-local features
    - dependency between consecutive rules
WordEdges Features (local)

<s> 布什 与 沙龙 举行 了 会谈 </s>

<s> Bush held a few talks </s>

- the first and last Chinese words in the rule
- the first and last English words in the rule
- the two Chinese words surrounding the rule
WordEdges Features (local)

- the first and last Chinese words in the rule
- the first and last English words in the rule
- the two Chinese words surrounding the rule
the first and last Chinese words in the rule

the first and last English words in the rule

the two Chinese words surrounding the rule
the first and last Chinese words in the rule
the first and last English words in the rule
the two Chinese words surrounding the rule
the first and last Chinese words in the rule
the first and last English words in the rule
the two Chinese words surrounding the rule
WordEdges Features (local)

<s> 布什 与 沙龙 举行 了 会谈 </s>

<s> Bush held a few talks </s>

• the first and last Chinese words in the rule
• the first and last English words in the rule
• the two Chinese words surrounding the rule

Combo Features:
WordEdges Features (local)

• the first and last Chinese words in the rule
• the first and last English words in the rule
• the two Chinese words surrounding the rule

Combo Features:
WordEdges Features (local)

<\text{s}> 布什 与 沙龙 举行了会谈 </\text{s}>

<\text{s}> Bush held a few talks </\text{s}>

- the first and last Chinese words in the rule
- the first and last English words in the rule
- the two Chinese words surrounding the rule

Combo Features:

100010 = 沙龙 | held
WordEdges Features (local)

<s> 布什 与 沙龙 </s> 举行 了 会谈 </s>

<s> Bush held a few talks </s>

- the first and last Chinese words in the rule
- the first and last English words in the rule
- the two Chinese words surrounding the rule

Combo Features:

100010=沙龙|held
the first and last Chinese words in the rule
the first and last English words in the rule
the two Chinese words surrounding the rule

Combo Features:
100010=沙龙|held
WordEdges Features (local)

- the first and last Chinese words in the rule
- the first and last English words in the rule
- the two Chinese words surrounding the rule

Combo Features:

100010 = 沙龙 | held
010001 = 举行 | talks
WordEdges Features (local)

• the first and last Chinese words in the rule
• the first and last English words in the rule
• the two Chinese words surrounding the rule

Combo Features:

100010 = 沙龙 | held
010001 = 举行 | talks
Lexical backoffs and combos

- Lexical features are often too sparse
- 6 kinds of lexical backoffs with various budgets
  - total budget can’t exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>52.9k 64.2k</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>characters</td>
<td>-</td>
<td>3.7k</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, full string</td>
<td>200</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td>6 8</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td>4 4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>POS tag</td>
<td>52 36</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>word type</td>
<td>-</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>
Lexical backoffs and combos

- Lexical features are often too sparse
- 6 kinds of lexical backoffs with various budgets
- Total budget can’t exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>52.9k 64.2k</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>characters</td>
<td>-</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, full</td>
<td>200</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td>6 8</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td>4 4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>POS tag</td>
<td>52 36</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>word type</td>
<td>-</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>
Lexical backoffs and combos

- Lexical features are often too sparse
- 6 kinds of lexical backoffs with various budgets
- Total budget can’t exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>-</td>
<td>52.9k</td>
<td>5</td>
</tr>
<tr>
<td>characters</td>
<td>-</td>
<td>3.7k</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, full string</td>
<td>200</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td>6</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>POS tag</td>
<td>-</td>
<td>52/36</td>
<td>2</td>
</tr>
<tr>
<td>word type</td>
<td>-</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

100010=沙龙|held
Lexical backoffs and combos

- Lexical features are often too sparse
- 6 kinds of lexical backoffs with various budgets
  - total budget can’t exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>held</td>
<td>52.9k</td>
<td>5</td>
</tr>
<tr>
<td>characters</td>
<td>a few</td>
<td>3.7k</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, full string</td>
<td>held</td>
<td>200</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td>held</td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td>held</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>POS tag</td>
<td>held</td>
<td>52</td>
<td>2</td>
</tr>
<tr>
<td>word type</td>
<td>held</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

100010=沙龙|held
P00010=NN|held
Lexical backoffs and combos

- Lexical features are often too sparse
- 6 kinds of lexical backoffs with various budgets
- total budget can’t exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>held</td>
<td>52.9k</td>
<td>5</td>
</tr>
<tr>
<td>characters</td>
<td></td>
<td>3.7k</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, full string</td>
<td></td>
<td>200</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td></td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td></td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>POS tag</td>
<td></td>
<td>52</td>
<td>2</td>
</tr>
<tr>
<td>word type</td>
<td></td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

100010=沙龙|held
P00010=NN|held
Lexical backoffs and combos

- Lexical features are often too sparse
- 6 kinds of lexical backoffs with various budgets
- Total budget can't exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>held</td>
<td>52.9k, 64.2k</td>
<td>5</td>
</tr>
<tr>
<td>characters</td>
<td>-</td>
<td>3.7k</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, full string</td>
<td>200</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td>6, 8</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td>4, 4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>POS tag</td>
<td>word type</td>
<td>4, 36</td>
<td>2</td>
</tr>
</tbody>
</table>

100010 = 沙龙 | held
P00010 = NN | held
010001 = 举行 | talks
Lexical backoffs and combos

Lexical features are often too sparse

6 kinds of lexical backoffs with various budgets

Total budget can’t exceed 10 (bilexical)

<table>
<thead>
<tr>
<th>Chinese</th>
<th>English</th>
<th>class size</th>
<th>budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>word</td>
<td>held</td>
<td>52.9k</td>
<td>5</td>
</tr>
<tr>
<td>characters</td>
<td>-</td>
<td>3.7k</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, full string</td>
<td></td>
<td>200</td>
<td>3</td>
</tr>
<tr>
<td>Brown cluster, prefix 6</td>
<td></td>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>Brown cluster, prefix 4</td>
<td></td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>POS tag</td>
<td></td>
<td>52</td>
<td>2</td>
</tr>
<tr>
<td>word type</td>
<td></td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

100010=沙龙|held
P00010=NN|held
010001=举行|talks
0c0001=举|talks
Non-Local Features (trivial)

• two consecutive rule ids (rule bigram model)
• the last two English words and the current rule
• should explore a lot more!
Non-Local Features (trivial)

- two consecutive rule ids (rule bigram model)
- the last two English words and the current rule
- should explore a lot more!
Non-Local Features (trivial)

<s> 布什 与 沙龙 举行 了 会谈 </s>

- two consecutive rule ids (rule bigram model)
- the last two English words and the current rule
- should explore a lot more!

<s> Bush held a few talks </s>
Experiments

- Date sets
## Experiments

- **Date sets**

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td>Ch-En</td>
<td>240k</td>
<td>nist06 news</td>
<td>newstest2012</td>
</tr>
<tr>
<td></td>
<td>Sp-En</td>
<td>170k</td>
<td>newstest2012</td>
<td>newtest2013</td>
</tr>
</tbody>
</table>
## Experiments

- **Date sets**

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td>Ch-En</td>
<td>240k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
</tbody>
</table>
Experiments

- Date sets

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td></td>
<td>240k</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent.</td>
<td>words</td>
</tr>
<tr>
<td>full</td>
<td>21.4%</td>
<td>8.8%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
<td>24.6%</td>
</tr>
</tbody>
</table>

Table 3: The ratio of sentence and word coverage on small and large training sets.
Experiments

- Date sets

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td>Ch-En</td>
<td>240k</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td>sent.</td>
<td>sent.</td>
</tr>
<tr>
<td>words</td>
<td>words</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td>full</td>
<td>21.4%</td>
<td>32.1%</td>
</tr>
<tr>
<td></td>
<td>8.8%</td>
<td>12.7%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
<td>67.3%</td>
</tr>
<tr>
<td></td>
<td>24.6%</td>
<td>32.8%</td>
</tr>
</tbody>
</table>

Table 3: The ratio of sentence and word coverage on small and large training sets.

10x dev
Experiments

- Date sets

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td></td>
<td>240k</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>small</th>
<th>large</th>
<th>sent.</th>
<th>words</th>
</tr>
</thead>
<tbody>
<tr>
<td>full</td>
<td></td>
<td>21.4%</td>
<td>8.8%</td>
</tr>
<tr>
<td>+prefix</td>
<td></td>
<td>61.3%</td>
<td>24.6%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>small</th>
<th>large</th>
<th>sent.</th>
<th>words</th>
</tr>
</thead>
<tbody>
<tr>
<td>full</td>
<td></td>
<td>32.1%</td>
<td>12.7%</td>
</tr>
<tr>
<td>+prefix</td>
<td></td>
<td>67.3%</td>
<td>32.8%</td>
</tr>
</tbody>
</table>

Table 3: The ratio of sentence and word coverage on small and large training data.

10x dev 120x dev
### Experiments

- **Date sets**

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td>Ch-En</td>
<td>240k</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td>Sp-En</td>
<td>170k</td>
<td>newstest2012</td>
<td>newstest2013</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent.</td>
<td>words</td>
</tr>
<tr>
<td>full</td>
<td>21.4%</td>
<td>8.8%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
<td>24.6%</td>
</tr>
</tbody>
</table>

**Table 3:** The ratio of sentence and word coverage on small and large training sets.
## Experiments

- **Date sets**

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td></td>
<td>240k</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td>Sp-En</td>
<td>170k</td>
<td>newstest2012</td>
<td>newstest2013</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>sent.</td>
<td>words</td>
</tr>
<tr>
<td>full</td>
<td>21.4%</td>
<td>8.8%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
<td>24.6%</td>
</tr>
</tbody>
</table>

Table 3: The ratio of sentence and word coverage on small and large training sets.
Experiments

- Date sets

<table>
<thead>
<tr>
<th>Scale</th>
<th>Language</th>
<th>sent.</th>
<th>dev</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Ch-En</td>
<td>30k</td>
<td>nist06 news</td>
<td>nist08 news</td>
</tr>
<tr>
<td>Large</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td>Sp-En</td>
<td>170k</td>
<td>newstest2012</td>
<td>newtest2013</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>small</th>
<th>large</th>
</tr>
</thead>
<tbody>
<tr>
<td>sent.</td>
<td>sent.</td>
</tr>
<tr>
<td>full</td>
<td>21.4%</td>
</tr>
<tr>
<td>+prefix</td>
<td>61.3%</td>
</tr>
</tbody>
</table>

Table 3: The ratio of sentence and word coverage on small and large training sets.
Perceptron: std, early, and max-violation

- standard perceptron (Liang et al’s “bold”) works poorly
  - b/c invalid update ratio is very high (search quality is low)
- max-violation converges faster than early update
Perceptron: std, early, and max-violation

- standard perceptron (Liang et al’s “bold”) works poorly
  - b/c invalid update ratio is very high (search quality is low)
- max-violation converges faster than early update

this explains why Liang et al ’06 failed
std ~ “bold”; local ~ “local”
Perceptron: std, early, and max-violation

- standard perceptron (Liang et al’s “bold”) works poorly
  - b/c invalid update ratio is very high (search quality is low)
- max-violation converges faster than early update

This explains why Liang et al ’06 failed
std ~ “bold”; local ~ “local”
Perceptron: std, early, and max-violation

- standard perceptron (Liang et al’s “bold”) works poorly
  - b/c invalid update ratio is very high (search quality is low)
- max-violation converges faster than early update

This explains why Liang et al ’06 failed
std ~ “bold”; local ~ “local”
Parallelized Perceptron

- mini-batch perceptron (Zhao and Huang, 2013) much faster than iterative parameter mixing (McDonald et al, 2010)
- 6 CPUs => ~4x speedup; 24 CPUs => ~7x speedup
Internal comparison with different features

- dense: 11 standard features for phrase-based MT
- ruleid: rule identification feature
- word-edges: word-edges features with back-offs
- non-local: non-local features with back-offs
Internal comparison with different features

- **dense**: 11 standard features for phrase-based MT
- **ruleid**: rule identification feature
- **word-edges**: word-edges features with back-offs
- **non-local**: non-local features with back-offs

![Graph showing BLEU scores for different features and iterations]
Internal comparison with different features

- **dense**: 11 standard features for phrase-based MT
- **ruleid**: rule identification feature
- **word-edges**: word-edges features with back-offs
- **non-local**: non-local features with back-offs

![Graph showing BLEU scores over iterations with different feature combinations.](image)

- **MERT**
- **ruleid**: 0.1%  
- **dense**: 11 features  
- **+0.9 bleu**
Internal comparison with different features

- **dense**: 11 standard features for phrase-based MT
- **ruleid**: rule identification feature
- **word-edges**: word-edges features with back-offs
- **non-local**: non-local features with back-offs

![Graph showing BLEU scores over iterations with different features](image)

- **MERT**:
  - wordedges: 99.6%  (+2.3)
  - ruleid: 0.1%  (+0.9 bleu)
- **dense**: 11 features
Internal comparison with different features

- **dense**: 11 standard features for phrase-based MT
- **ruleid**: rule identification feature
- **word-edges**: word-edges features with back-offs
- **non-local**: non-local features with back-offs

![Graph showing BLEU scores over iterations with various feature sets.](image)

- dense: 11 features
- ruleid: 0.1%
- word-edges: 99.6%
- non-local: 0.3%

BLEU scores for different features:
- MERT + non-local: +0.7
- MERT + word-edges: +2.3
- MERT + ruleid: +0.9 bleu
- MERT + dense: 26.4
External comparison with MERT & PRO

- MERT, PRO-dense/medium/sparse all tune on dev-set
- PRO-sparse use the same feature as ours
Final Results on FBIS Data

- Moses: state-of-the-art phrase-based system in C++
- Cubit: phrase-based system (Huang and Chiang, 2007) in python
  - almost identical baseline scores with MERT
  - max-violation takes ~47 hours on 24 CPUs (23M features)
Final Results on FBIS Data

- **Moses**: state-of-the-art phrase-based system in C++
- **Cubit**: phrase-based system ([Huang and Chiang, 2007](#)) in python
  - almost identical baseline scores with MERT
- max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
</tbody>
</table>
## Final Results on FBIS Data

- **Moses**: state-of-the-art phrase-based system in C++
- **Cubit**: phrase-based system ([Huang and Chiang, 2007](#)) in python
  - almost identical baseline scores with MERT
  - max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>PRO</td>
<td>dev set</td>
<td>11</td>
<td>25.6</td>
<td>22.6</td>
</tr>
</tbody>
</table>
Final Results on FBIS Data

- Moses: state-of-the-art phrase-based system in C++
- Cubit: phrase-based system (Huang and Chiang, 2007) in python
- almost identical baseline scores with MERT
- max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>PRO</td>
<td>dev set</td>
<td>11</td>
<td>25.6</td>
<td>22.6</td>
</tr>
<tr>
<td></td>
<td>PRO</td>
<td></td>
<td>3k</td>
<td>26.3</td>
<td>23.0</td>
</tr>
</tbody>
</table>
Final Results on FBIS Data

- **Moses**: state-of-the-art phrase-based system in C++
- **Cubit**: phrase-based system ([Huang and Chiang, 2007](#)) in python
  - almost identical baseline scores with MERT
  - max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11 11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>MERT</td>
<td>dev set</td>
<td>11 11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>PRO</td>
<td>dev set</td>
<td>11 11</td>
<td>25.6</td>
<td>22.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3k 36k</td>
<td>26.3</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>17.7</td>
<td>14.3</td>
</tr>
</tbody>
</table>
Final Results on FBIS Data

- Moses: state-of-the-art phrase-based system in C++
- Cubit: phrase-based system (Huang and Chiang, 2007) in python
- almost identical baseline scores with MERT
- max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>PRO</td>
<td>dev set</td>
<td>11</td>
<td>25.6</td>
<td>22.6</td>
</tr>
<tr>
<td></td>
<td>PRO</td>
<td>dev set</td>
<td>3k</td>
<td>26.3</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td>PRO</td>
<td>dev set</td>
<td>36k</td>
<td>17.7</td>
<td>14.3</td>
</tr>
<tr>
<td>MaxForce</td>
<td>Train set</td>
<td>23M</td>
<td>27.8</td>
<td>24.5</td>
<td></td>
</tr>
</tbody>
</table>
## Final Results on FBIS Data

- **Moses**: state-of-the-art phrase-based system in C++
- **Cubit**: phrase-based system ([Huang and Chiang, 2007](#)) in python
  - almost identical baseline scores with MERT
  - max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>PRO</td>
<td>dev set</td>
<td>11</td>
<td>25.6</td>
<td>22.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3k</td>
<td>26.3</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>36k</td>
<td>17.7</td>
<td>14.3</td>
</tr>
<tr>
<td>MaxForce</td>
<td></td>
<td>Train set</td>
<td>23M</td>
<td>27.8</td>
<td>24.5</td>
</tr>
</tbody>
</table>

\[+2.3\]
Final Results on FBIS Data

- Moses: state-of-the-art phrase-based system in C++
- Cubit: phrase-based system (Huang and Chiang, 2007) in python
  - almost identical baseline scores with MERT
- max-violation takes ~47 hours on 24 CPUs (23M features)

<table>
<thead>
<tr>
<th>System</th>
<th>Alg.</th>
<th>Tune on</th>
<th>Features</th>
<th>Dev</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.5</td>
<td>22.5</td>
</tr>
<tr>
<td></td>
<td>MERT</td>
<td>dev set</td>
<td>11</td>
<td>25.4</td>
<td>22.5</td>
</tr>
<tr>
<td>Cubit</td>
<td>PRO</td>
<td>dev set</td>
<td>11</td>
<td>25.6</td>
<td>22.6</td>
</tr>
<tr>
<td></td>
<td>PRO</td>
<td>dev set</td>
<td>3k</td>
<td>26.3</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td>PRO</td>
<td>dev set</td>
<td>36k</td>
<td>17.7</td>
<td>14.3</td>
</tr>
<tr>
<td>MaxForce</td>
<td>Train set</td>
<td>23M</td>
<td>27.8</td>
<td>24.5</td>
<td>+2.3</td>
</tr>
</tbody>
</table>
Results on Spanish-English set

- Data-set: Europarl corpus, 170k sentences
- dev/test set: newtest2012 / 2013 (one-reference only)
  - +1 in 1-ref bleu ~ +2 in 4-ref bleu
  - bleu improvement is comparable to Chinese w/ 4-refs

<table>
<thead>
<tr>
<th>system</th>
<th>algorithm</th>
<th>#feat.</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>Mert</td>
<td>11</td>
<td>27.4</td>
<td>24.4</td>
</tr>
<tr>
<td>Cubit</td>
<td>MaxForce</td>
<td>21M</td>
<td><strong>28.7</strong></td>
<td><strong>25.5</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sp-En</th>
<th>sent.</th>
<th>word.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reachable ratio</td>
<td>55%</td>
<td>43.9%</td>
</tr>
</tbody>
</table>
Results on Spanish-English set

- Data-set: Europarl corpus, 170k sentences
- dev/test set: newtest2012 / 2013 (one-reference only)
  - +1 in 1-ref bleu ~ +2 in 4-ref bleu
  - bleu improvement is comparable to Chinese w/ 4.refs

<table>
<thead>
<tr>
<th>system</th>
<th>algorithm</th>
<th>#feat.</th>
<th>dev</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moses</td>
<td>Mert</td>
<td>11</td>
<td>27.4</td>
<td>24.4</td>
</tr>
<tr>
<td>Cubit</td>
<td>MaxForce</td>
<td>21M</td>
<td><strong>28.7</strong></td>
<td><strong>25.5</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sp-En</th>
<th>sent.</th>
<th>word.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reachable ratio</td>
<td>55%</td>
<td>43.9%</td>
</tr>
</tbody>
</table>
Conclusion

- a simple yet effective online learning approach for MT
- scaled to (a large portion of) the training set for the first time
- able to incorporate 20M sparse lexicalized features
- no need to define BLEU+1, or hope/fear derivations
- no learning rate or hyperparameters
- +2.3/+2.0 BLEU points better than MERT/PRO
- the three ingredients that made it work
  - violation-fixing perceptron: early-update and max-violation
  - forced decoding lattice helps
  - minibatch parallelization scales it up to big data
Roadmap of the techniques

structured perceptron (Collins, 2002)
Roadmap of the techniques

structured perceptron (Collins, 2002)

latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)
Roadmap of the techniques

- **Structured perceptron** (Collins, 2002)
  - Latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)
  - Perceptron w/ inexact search (Collins & Roark, 2004; Huang et al. 2012)
Roadmap of the techniques

structured perceptron (Collins, 2002)

- latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)
- perceptron w/ inexact search (Collins & Roark, 2004; Huang et al., 2012)
- latent-variable perceptron w/ inexact search (Yu et al., 2013)
Roadmap of the techniques

structured perceptron (Collins, 2002)

latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)

perceptron w/ inexact search (Collins & Roark, 2004; Huang et al 2012)

latent-variable perceptron w/ inexact search (Yu et al 2013)

hiero syntactic parsing semantic parsing transliteration
Roadmap of the techniques

- structured perceptron (Collins, 2002)
- latent-variable perceptron (Zettlemoyer and Collins, 2005; Sun et al., 2009)
- perceptron w/ inexact search (Collins & Roark, 2004; Huang et al. 2012)
- latent-variable perceptron w/ inexact search (Yu et al. 2013)

Replacing EM for partially-observed data

- hiero
- syntactic parsing
- semantic parsing
- transliteration
20 years of Statistical MT

- word alignment: IBM models (Brown et al 90, 93)
- translation model (choose one from below)
  - SCFG (ITG: Wu 95, 97; Hiero: Chiang 05, 07) or STSG (GHKM 04, 06; Liu+ 06; Huang+ 06)
  - PBMT (Och+Ney 02; Koehn et al 03)
- evaluation metric: BLEU (Papineni et al 02)
- decoding algorithm: cube pruning (Chiang 07; Huang+Chiang 07)
- training algorithm (choose one from below)
  - MERT (Och 03): ~10 dense features on dev set
  - MIRA (Chiang et al 08-12) or PRO (Hopkins+May 11): ~10k feats on dev set
  - MaxForce: 20M+ feats on training set; +2/+1.5 BLEU over MERT/PRO
    - Max-Violation Perceptron with Forced Decoding: fixes search errors
    - first successful effort of online large-scale discriminative training for MT
When learning with vastly inexact search, you should use a principled method such as max-violation.

Thank you!