Confidence-based Rewriting of Machine Translation Output

Benjamin Marie[1,2]   Aurélien Max[1,3]

(1) LIMSI-CNRS   (2) Lingua et Machina   (3) Université Paris-Sud

Introduction

- Phrase-Based Statisical Machine Translation (PBSMT) systems use many features during decoding to assess the quality of translation hypotheses
- For other features, several difficulties of integration to overcome, e.g. :
  - **need of a complete hypothesis**
    e.g. sentence-level syntactic features
  - **computational cost**
    e.g. Neural Network language models
  - **need of a first decoding**
    e.g. *a posteriori* confidence models
- How to use such features *efficiently* in PBSMT ?

Reranking of translation hypotheses

A solution

- rerank the *n*-best list of the decoder using new, complex features
- can achieve good performance with some features
  (Och et al., 2004; Carter and Monz, 2011; Le et al., 2012; Luong et al., 2014)

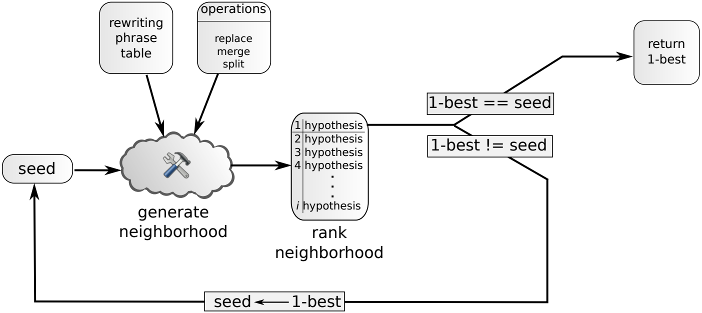
2 strong limitations

- lack of diversity (Gimpel et al., 2013)
- inherit a limited selection of hypotheses made by the decoder

A rewriting system

A `rewriter` to extend the exploration

- idea: search for new promising hypotheses **not** in the *n*-best list

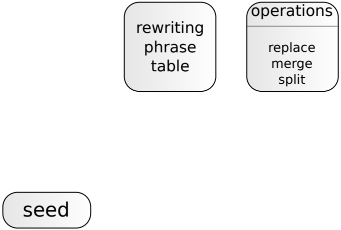


The seed: an hypothesis to rewrite

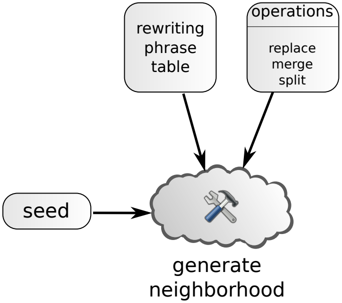
seed

A rewriting phrase table

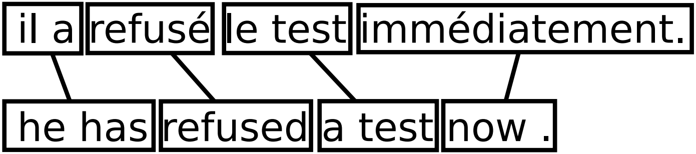
A set of rewriting operations



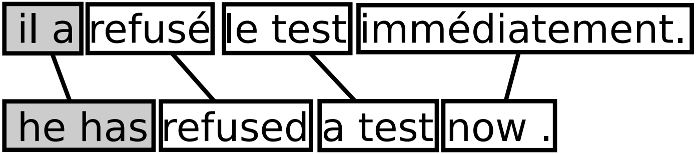
Neighborhood generation



Neighborhood generation : `replace`



Neighborhood generation : `replace`



Neighborhood generation : `replace`

il a | refusé | le test | immédiatement.

he has | refused | a test | now .

he has refused a test now .

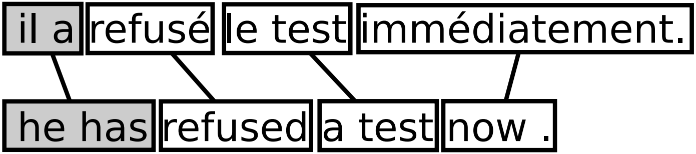
he refused a test now .

he had refused a test now .

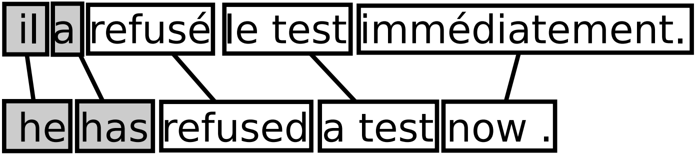
it has refused a test now .

it refused a test now .

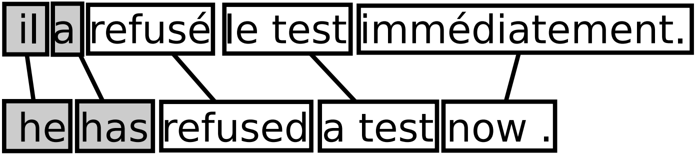
Neighborhood generation : `split`



Neighborhood generation : `split`

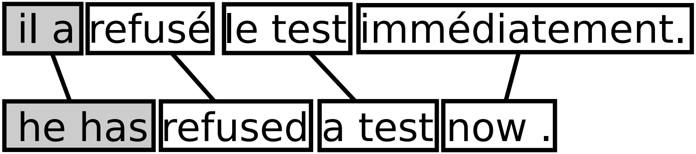


Neighborhood generation : `split`

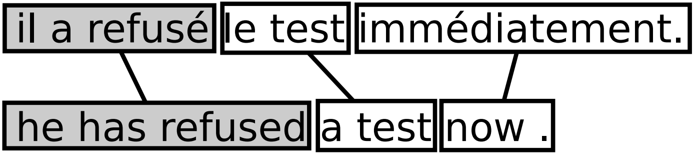


he has refused a test now .
he is refused a test now .
he had refused a test now .
it has refused a test now .
it have refused a test now .

Neighborhood generation : `merge`



Neighborhood generation : `merge`



Neighborhood generation : `merge`

il a refusé | le test | immédiatement.

he has refused | a test | now .

he has refused a test now .

he refused a test now .

he rejected a test now .

he has just refused a test now .

he has a test now .

Rewriting phrase table

Building the rewriting table

- **Method 1**: take the *i* **best translations** according to $\mathbf{p(e|f)}$
- **Method 2**: take the **bi-phrases appearing in the decoder *k*-best list**

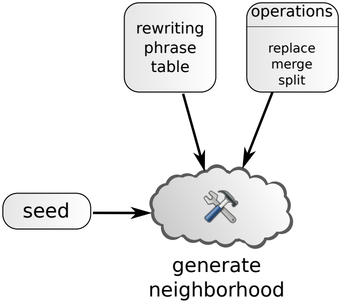
Method 1

- produces very large neighborhoods
- not suitable for costly features

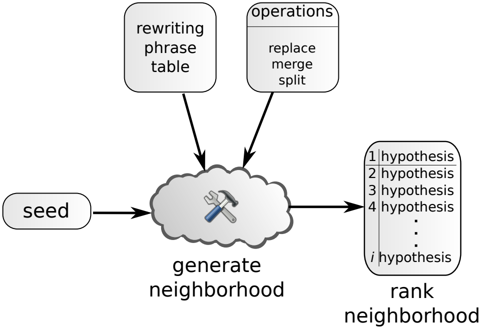
Method 2

- produces **very small** and **adapted** rewriting phrase table for **each sentence**
- keeps only bi-phrases for which the decoder was **the most confident**

Neighborhood generation



Ranking of the neighborhood



Ranking of the neighborhood

Objective

- rank (manageable) neighborhoods using complex features

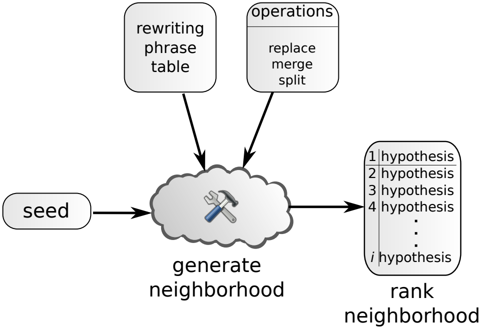
Training the reranker: 2 kinds of examples

- *n*-best produced by the decoder
- neighborhoods produced by one iteration of `rewriter`

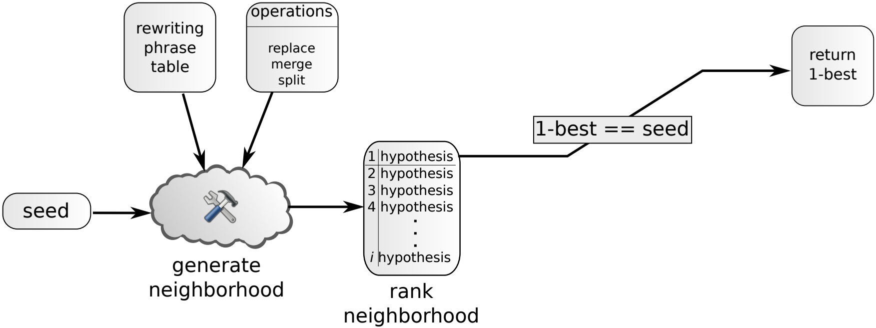
Training algorithm

- `kb-mira` (Cherry and Foster, 2012)

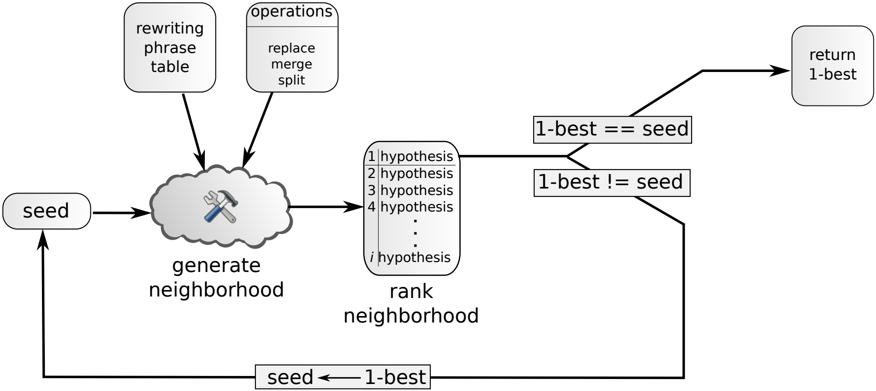
Ranking of the neighborhood



Greedy search



Greedy search



Greedy search

- greedy search algorithm for PBSMT (Langlais et al., 2007)
  - choose at each iteration the best rewriting/operation according to the (new) scoring function

Source	il a refusé le test immédiatement .
Reference	he refused the test straight away .
seed	$\boxed{\text{il a}_1}$ $\boxed{\text{refusé}_2}$ $\boxed{\text{le test}_3}$ $\boxed{\text{immédiatement .}_4}$
↓	$\boxed{\text{he has}_1}$ $\boxed{\text{refused}_2}$ $\boxed{\text{a test}_3}$ $\boxed{\text{now .}_4}$

Greedy search

- greedy search algorithm for PBSMT (Langlais et al., 2007)
  - choose at each iteration the best rewriting/operation according to the (new) scoring function

Source	il a refusé le test immédiatement .
Reference	he refused the test straight away .
seed	$\boxed{\text{il a}_1}$ $\boxed{\text{refusé}_2}$ $\boxed{\text{le test}_3}$ $\boxed{\text{immédiatement .}_4}$
↓	$\boxed{\text{he has}_1}$ $\boxed{\text{refused}_2}$ $\boxed{\text{a test}_3}$ $\boxed{\text{now .}_4}$
`merge`	$\boxed{\text{il a refusé}_1}$ $\boxed{\text{le test}_2}$ $\boxed{\text{immédiatement .}_3}$
iteration 1	$\boxed{\text{he refused}_1}$ $\boxed{\text{a test}_2}$ $\boxed{\text{now .}_3}$

Greedy search

- greedy search algorithm for PBSMT (Langlais et al., 2007)
  - choose at each iteration the best rewriting/operation according to the (new) scoring function

Source	il a refusé le test immédiatement .
Reference	he refused the test straight away .
seed	[il a]$_1$ [refusé]$_2$ [le test]$_3$ [immédiatement .]$_4$
↓	[he has]$_1$ [refused]$_2$ [a test]$_3$ [now .]$_4$
`merge`	[il a refusé]$_1$ [le test]$_2$ [immédiatement .]$_3$
iteration 1	[he refused]$_1$ [a test]$_2$ [now .]$_3$
`split`	[il a refusé]$_1$ [le test]$_2$ [immédiatement]$_3$ [.]$_4$
iteration 2	[he refused]$_1$ [a test]$_2$ [straight away]$_3$ [.]$_4$

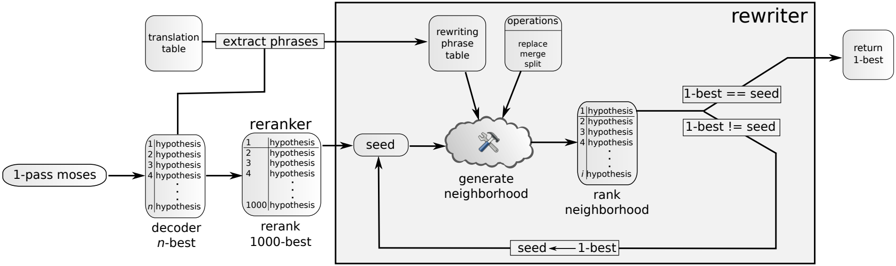
Greedy search

- greedy search algorithm for PBSMT (Langlais et al., 2007)
  - choose at each iteration the best rewriting/operation according to the (new) scoring function

Source	il a refusé le test immédiatement .
Reference	he refused the test straight away .
seed	$\boxed{\text{il a}_1}$ $\boxed{\text{refusé}_2}$ $\boxed{\text{le test}_3}$ $\boxed{\text{immédiatement .}_4}$
↓	$\boxed{\text{he has}_1}$ $\boxed{\text{refused}_2}$ $\boxed{\text{a test}_3}$ $\boxed{\text{now .}_4}$
`merge`	$\boxed{\text{il a refusé}_1}$ $\boxed{\text{le test}_2}$ $\boxed{\text{immédiatement .}_3}$
iteration 1	$\boxed{\text{he refused}_1}$ $\boxed{\text{a test}_2}$ $\boxed{\text{now .}_3}$
`split`	$\boxed{\text{il a refusé}_1}$ $\boxed{\text{le test}_2}$ $\boxed{\text{immédiatement}_3}$ $\boxed{\text{.}_4}$
iteration 2	$\boxed{\text{he refused}_1}$ $\boxed{\text{a test}_2}$ $\boxed{\text{straight away}_3}$ $\boxed{\text{.}_4}$
`replace`	$\boxed{\text{il a refusé}_1}$ $\boxed{\text{le test}_2}$ $\boxed{\text{immédiatement}_3}$ $\boxed{\text{.}_4}$
iteration 3	$\boxed{\text{he refused}_1}$ $\boxed{\text{the test}_2}$ $\boxed{\text{straight away}_3}$ $\boxed{\text{.}_4}$

Experiments

The whole framework



Experimental settings

- **translation tasks**: English↔French
  - `Ted Talks`
  - `WMT'14 medical`
  - `WMT'12`
- **baseline systems**
  - `Moses` PBSMT (Koehn et al., 2007)
  - `kb-mira` reranker using all the features below
- **features**
  - decoder features : all the features used by the 1st-pass decoder
  - neural network models : 10-gram monolingual (Le et al., 2011) and bilingual (Le et al., 2012) `SOUL` models
  - Part-of-speech language model: 6-gram model
  - IBM1 scores
  - phrase posterior probabilities

Results

Task	system	en-fr BLEU	en-fr Δ	fr-en BLEU	fr-en Δ
WMT'12	1-pass Moses	31.8		29.4	
	reranker	32.9	+1.1	30.3	+0.9
TED Talks	1-pass Moses	32.3		32.5	
	reranker	32.8	+0.5	33.0	+0.5
WMT'14 medical	1-pass Moses	38.3			
	reranker	41.8	+3.5		

⇒ moderate (TED Talks) to strong (medical) improvements with reranker over the 1st-pass decoder

Results

Task	system	en-fr BLEU	en-fr Δ	fr-en BLEU	fr-en Δ
WMT'12	1-pass Moses	31.8		29.4	
	reranker	32.9	+1.1	30.3	+0.9
	rewriter	33.5	+1.7	30.8	+1.4
TED Talks	1-pass Moses	32.3		32.5	
	reranker	32.8	+0.5	33.0	+0.5
	rewriter	33.7	+1.4	33.4	+0.9
WMT'14 medical	1-pass Moses	38.3			
	reranker	41.8	+3.5		
	rewriter	43.4	+5.1		

$\Rightarrow$ `rewriter` increases by ∼50% the `reranker` improvement

Results

Task	system	en-fr BLEU	en-fr Δ	fr-en BLEU	fr-en Δ
WMT'12	1-pass Moses	31.8		29.4	
	reranker	32.9		30.3	
	rewriter	33.5	+0.6	30.8	+0.5
TED Talks	1-pass Moses	32.3		32.5	
	reranker	32.8		33.0	
	rewriter	33.7	+0.9	33.4	+0.4
WMT'14 medical	1-pass Moses	38.3			
	reranker	41.8			
	rewriter	43.4	+1.6		

⇒ `rewriter` increases by ∼50% the `reranker` improvement

Analysis: outline

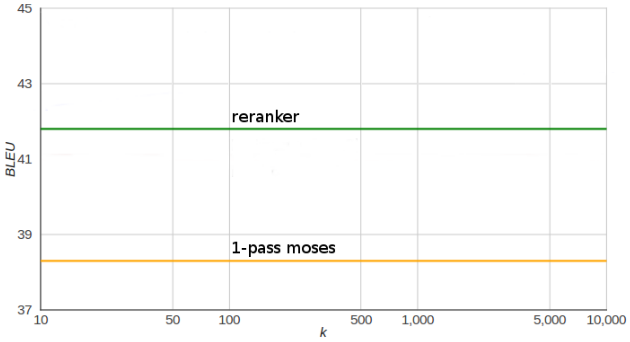
1. training procedure
2. rewriting phrase table
3. best attainable performance
4. performance depending on translation quality
5. sentence-level performance
6. other findings

Training examples

	dev BLEU	test BLEU	Δ
`reranker`	44.1	41.8	
`rewriter` training			
`1-pass Moses` 1,000-best	44.1	39.2	-2.6
`rewriter` neighborhoods	44.5	**43.4**	**+1.6**

⇒ `rewriter` **must** be trained on `rewriter` neighborhoods

Rewriting phrase table performance



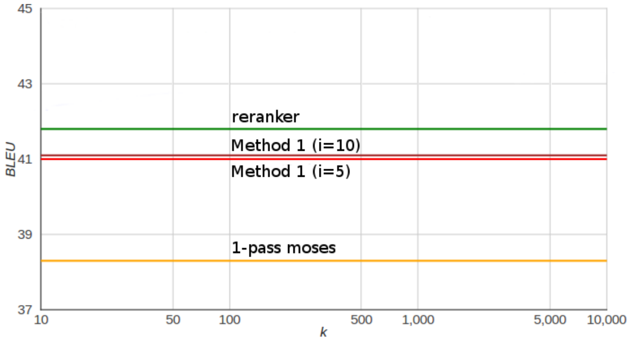
Method 1: extraction according to $p(e|f)$

- damages `reranker` output

Method 2: extraction from a k-best list

- improvements for all tested k, even for small values (best for k = 10,000)

Rewriting phrase table performance



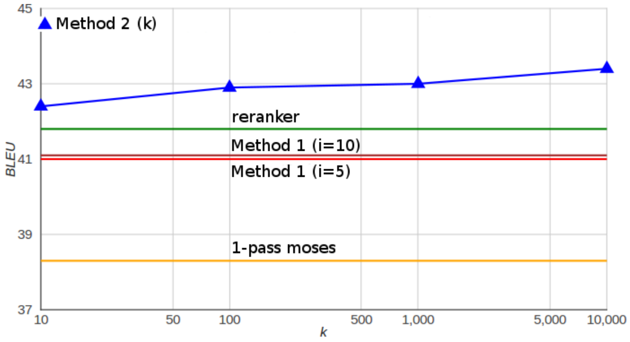
Method 1: extraction according to $p(e|f)$

- damages `reranker` output

Method 2: extraction from a k-best list

- improvements for all tested k, even for small values (best for $k = 10{,}000$)

Rewriting phrase table performance



Method 1: extraction according to $p(e|f)$

- damages `reranker` output

Method 2: extraction from a k-best list

- improvements for all tested k, even for small values (best for k = 10,000)

Rewriting phrase table size

rewriting phrase table		unique bi-phrases	Δ-BLEU w.r.t. `reranker`
Method 1	i = 5	85,530	-0.8
	i = 10	149,887	-0.7
Method 2	k = 10	21,398	+0.6
	k = 100	28,730	+1.1
	k = 1,000	33,929	+1.2
	k = 10,000	38,455	+1.6

- compact phrase tables when extracted from k-best lists (Method 2)
- much larger when extracted according to $p(e|f)$ (Method 1)

Best attainable performance

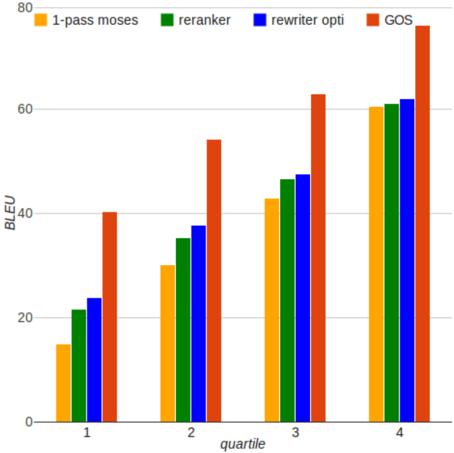
- Greedy Oracle Search (GOS) (Marie and Max, 2013)
  - make the best local decision at each iteration
  - use sentence-BLEU as scoring function

baseline		test BLEU	Δ
`reranker`		41.8	
rewriting phrase table			
method 1	i = 5	50.6	+8.8
	i = 10	54.5	+12.7
method 2	k = 10	45.9	+4.1
	k = 100	50.2	+8.4
	k = 1,000	53.3	+11.5
	k = 10,000	**58.7**	+16.9

⇒ strong oracle improvements, even for compact rewriting tables

⇒ extracting from k-best lists much more promising

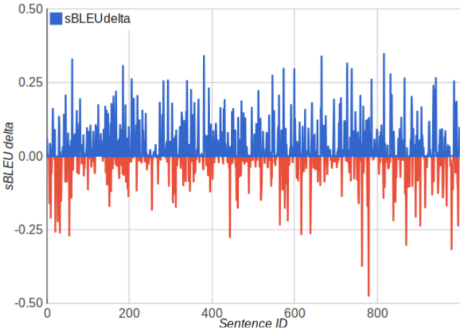
Performance depending on translation quality



- `rewriter` improvement :
  - quartile 4 : +1.4 BLEU
  - quartile 1 : +9.0 BLEU

⇒ larger improvements on bad/difficult translations

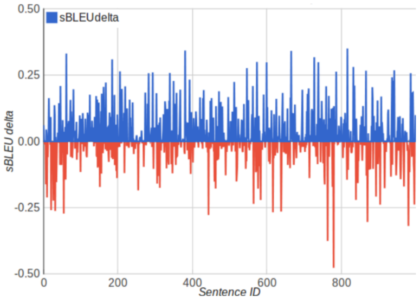
Sentence-level performance



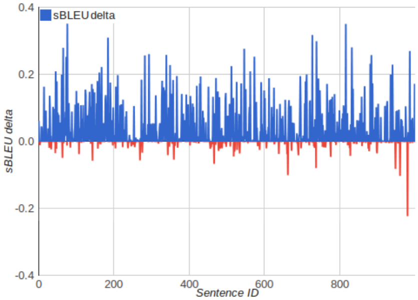
- according to sentence-BLEU, after rewriting :
  - 40.8% better
  - 29.2% worse
  - 30% unchanged

⇒ large room for further improvement

Sentence-level performance: semi-oracle experiment



(a) automatic rewriting



(b) semi-oracle rewriting

- protecting the phrases appearing in the reference translation: +1.5 BLEU

⇒ strong value of better confidence estimates

Other findings

1. 70% of **new** hypotheses **not** in `1-pass Moses` 1,000-best
2. on average (only) 116 hypotheses per sentence in the neighborhood
3. searching using a **beam** of size 10: 1.6 $\rightarrow$ 1.9 BLEU
4. manual evaluation revealed both **fluency** and **accuracy** improvements

Conclusion

- an **efficient** and **simple** procedure to make a **better use of features** difficult to integrate during decoding
- **produces useful hypotheses not in the decoder *n*-best list**
- relies on the **decoder confidence** to extract the rewriting rules
- improvements on **3 different tasks** and **2 language directions** over a reranked baseline **using the same features**

Future work

- exploit more features : lexical-coherence (Hardmeier et al., 2012), syntactic features (Post, 2011), word posterior probabiliy (Ueffing and Ney, 2007), etc.
- identify correct phrases to protect them from rewriting
- adapt `rewriter`'s objective function to the sentence
- use a `paraphrase` operation rewriting the source sentence to produce new target phrases (Marie and Max, 2013)
- use automatic alternative reference translations (Madnani and Dorr, 2013)
- use `rewriter` in interaction with human translators

Future work

- exploit more features : lexical-coherence (Hardmeier et al., 2012), syntactic features (Post, 2011), word posterior probabiliy (Ueffing and Ney, 2007), etc.
- identify correct phrases to protect them from rewriting
- adapt `rewriter`'s objective function to the sentence
- use a `paraphrase` operation rewriting the source sentence to produce new target phrases (Marie and Max, 2013)
- use automatic alternative reference translations (Madnani and Dorr, 2013)
- use `rewriter` in interaction with human translators

Future work

- exploit more features : lexical-coherence (Hardmeier et al., 2012), syntactic features (Post, 2011), word posterior probabiliy (Ueffing and Ney, 2007), etc.
- identify correct phrases to protect them from rewriting
- adapt `rewriter`'s objective function to the sentence
- use a `paraphrase` operation rewriting the source sentence to produce new target phrases (Marie and Max, 2013)
- use automatic alternative reference translations (Madnani and Dorr, 2013)
- use `rewriter` in interaction with human translators

Future work

- exploit more features : lexical-coherence (Hardmeier et al., 2012), syntactic features (Post, 2011), word posterior probabiliy (Ueffing and Ney, 2007), etc.
- identify correct phrases to protect them from rewriting
- adapt `rewriter`'s objective function to the sentence
- use a `paraphrase` operation rewriting the source sentence to produce new target phrases (Marie and Max, 2013)
- use automatic alternative reference translations (Madnani and Dorr, 2013)
- use `rewriter` in interaction with human translators

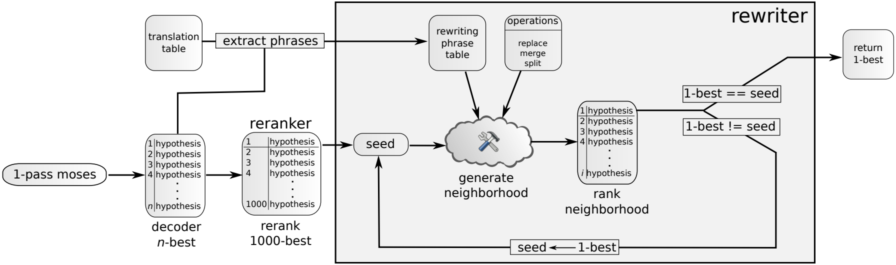
Future work

- exploit more features : lexical-coherence (Hardmeier et al., 2012), syntactic features (Post, 2011), word posterior probabiliy (Ueffing and Ney, 2007), etc.
- identify correct phrases to protect them from rewriting
- adapt `rewriter`'s objective function to the sentence
- use a `paraphrase` operation rewriting the source sentence to produce new target phrases (Marie and Max, 2013)
- use automatic alternative reference translations (Madnani and Dorr, 2013)
- use `rewriter` in interaction with human translators

Future work

- exploit more features : lexical-coherence (Hardmeier et al., 2012), syntactic features (Post, 2011), word posterior probabiliy (Ueffing and Ney, 2007), etc.
- identify correct phrases to protect them from rewriting
- adapt `rewriter`'s objective function to the sentence
- use a `paraphrase` operation rewriting the source sentence to produce new target phrases (Marie and Max, 2013)
- use automatic alternative reference translations (Madnani and Dorr, 2013)
- use `rewriter` in interaction with human translators

Thanks for listening !
Questions ?



Confidence-based Rewriting of Machine Translation Output

Benjamin Marie & Aurélien Max

emnlp$_{2014}$

Carter, S. and Monz, C. (2011). Syntactic discriminative language model rerankers for statistical machine translation. Machine Translation.

Cherry, C. and Foster, G. (2012). Batch Tuning Strategies for Statistical Machine Translation. In Proceedings of NAACL, Montréal, Canada.

de Gispert, A., Blackwood, G., Iglesias, G., and Byrne, W. (2012). N-gram posterior probability confidence measures for statistical machine translation: an empirical study. Machine Translation.

Gimpel, K., Batra, D., Dyer, C., Shakhnarovich, G., and Tech, V. (2013). A Systematic Exploration of Diversity in Machine Translation. In Proceedings of EMNLP 2013, Seatlle, USA.

Hardmeier, C., Nivre, J., and Tiedeman, J. (2012). Document-Wide Decoding for Phrase-Based Statistical Machine Translation. In Proceedings of EMNLP, Jeju Island, Korea.

Koehn, P., Hoang, H., Birch, A., Callison-burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open Source Toolkit for Statistical Machine Translation. In Proceedings of ACL, demos, Prague, Czech Republic.

Langlais, P., Patry, A., and Gotti, F. (2007). A Greedy Decoder for Phrase-Based Statistical Machine Translation. In Proceedings of Conference on Theoretical and Methodological Issues in Machine Translation (TMI), Skovde, Sweden.

Le, H.-S., Allauzen, A., and Yvon, F. (2012). Continuous Space Translation Models with Neural Networks. In Proceedings of NAACL, Montréal, Canada.

Le, H.-S., Oparin, I., Allauzen, A., Gauvain, J.-L., and Yvon, F. (2011). Structured Output Layer Neural Network Language Model. In Proceedings of ICASSP, Prague, Czech Republic.

Luong, N.-Q., Besacier, L., and Lecouteux, B. (2014). Word Confidence Estimation for SMT N -best List Re-ranking. In Proceedings of the Workshop on Humans and Computer-assisted Translation (HaCaT), Gothenburg, Sweden.

Madnani, N. and Dorr, B. J. (2013). Generating Targeted Paraphrases for Improved Translation. ACM Transactions on Intelligent Systems and Technology, special issue on Paraphrasing, 4(3).

Marie, B. and Max, A. (2013). A Study in Greedy Oracle Improvement of Translation Hypotheses. In Proceedings of IWSLT, Heidelberg, Germany.

Och, F. J., Gildea, D., Khudanpur, S., Sarkar, A., Yamada, K., Fraser, A., Kumar, S., Shen, L., Smith, D., Eng, K., Jain, V., Jin, Z., and Radev, D. (2004). A Smorgasbord of Features for Statistical Machine Translation. In Proceedings of NAACL, Boston, USA.

Post, M. (2011). Judging Grammaticality with Tree Substitution Grammar Derivations. In Proceedings of ACL, short papers, Portland, USA.

Ueffing, N. and Ney, H. (2007). Word-Level Confidence Estimation for Machine Translation. Computational Linguistics.