Non-linear n-best List Reranking with Few Features

Artem Sokolov, **Guillaume Wisniewski**, François Yvon

Université Paris Sud and LIMSI–CNRS

October 31, 2012

Outline

Motivation

Model

Experiments

Conclusion

Performance Discrepancy in SMT (1)

Anatomy of a SMT system:

1. build a (large) search space of hypotheses translation
2. define a linear-scoring function
   - linear combination of $\simeq 20$ features
   - weights are chosen to maximize BLEU score on a dev set (MERT)
3. look for the highest-scoring hypothesis (MAP inference)

Research in SMT:

- change any of the previous point...
- and be happy with a 0.5-1 BLEU point improvement...
- ...until we search for oracle hypotheses

Performance Discrepancy in SMT (2)

Oracle decoding [Wisniewski 10, Sokolov 12]

- failure analysis procedure
- use knowledge of the reference to guide search during decoding
- find the "best" hypotheses (i.e.: highest BLEU score achievable)

	found by decoder	lattice oracles
BLEU fr $\rightarrow$ en	~ 28	~ 50
BLEU de $\rightarrow$ en	~ 22	~ 38
BLEU en $\rightarrow$ de	~ 16	~ 30

$\Rightarrow$ potentially **two-fold** improvement

How to Solve the Performance Discrepancy Problem?

- oracles not reachable even with “advanced” learning:
  - lattice MERT [Macherey 08, Kumar 09, Sokolov 11]
  - exact MERT [Galley 11]
  - MIRA [Chiang 08]
  - tuning as ranking [Hopkins 11]
- adding more features has only limited impact
  - e.g.: +1,5 BLEU with 11,001 features [Chiang 09]
- **is scoring function main bottleneck?**
  - poor and few features?
  - wrong models? ←**this presentation**

Goal of this work:

Can conventional SMT systems benefit from non-linear scoring?

More Precisely

Goal: first attempt to assess the impact of using a non-linear scoring function

First attempt:

- n-best re-ranking to avoid a tight integration with the decoder
- only consider the standard features used by a Moses system

Reranking Model

"Classical reranking" model:

1. Training:
   - run full training (MERT)
   - take last iteration's n-best lists
   - train the re-scoring function
2. Testing:
   - generate n-best lists
   - score all hypotheses with the non-linear function
   - select the best scoring hypothesis

Non-linear Scoring Function

"New" scoring function:

$$H(\mathbf{e}, \mathbf{f}) = \sum_{t=1}^{T} \alpha_t \cdot h_t\left(\bar{g}(\mathbf{e}, \mathbf{f})\right)$$

where:

- $\bar{g}(\mathbf{e}, \mathbf{f})$ feature vector
- α_t weights
- h_t "simple" non-linear functions (weak-learner)

$\Rightarrow$ class of functions considered by boosting algorithms

Learning Criterion

Loss function

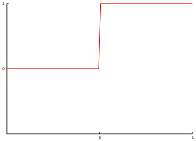
- hypotheses naturally ordered under sentence-level BLEU score
- ensure that two sentences are ordered in the same order according to their score and their sentence-level BLEU approximation
- deduce parameters comparing even mediocre or bad hypothesis

Tuning with Ranking

- first introduced by [Hopkins 11]
- earlier ranking approaches redefined losses, not scoring functions

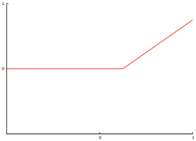
Hyper-Parameters

1. Number of Components
   - $T =$ number of weak learners to combine
2. Weak Functions
   - one weak learner per features



decision stumps:

- simplest weak-learner
- state-of-the-art performance in many tasks



piece-wise linear learners

- number of pieces is chosen automatically
- linear as a special case

Experimental Setup (1)

Decoder

- NCode: in-house phrase-based decoder
- similar results with Moses

Two Configurations

- **basic: 11** features (found in any decoder), **100**-best
  - language model
  - distortion and reordering models,
  - translation model (lexicalized)
  - words and phrases penalties
- **extended: 23** features (WMT'12 best system for fr$\leftrightarrow$en), **300**-best
  - lexicalized reordering models
  - add neural-network models features (LM & TM)

Experimental Setup (2)

Datasets

All experiments were done on the WMT data

- WMT'09 for training (both MERT & RankBoost)
- test on WMT'10, WMT'11 and WMT'12

MERT setup

- MERT is unstable $\Rightarrow$ 8 independent (re)runs, each with:
  - 20 init. points restarts
  - 30 random direction (additional to axes)

Feature Transformations

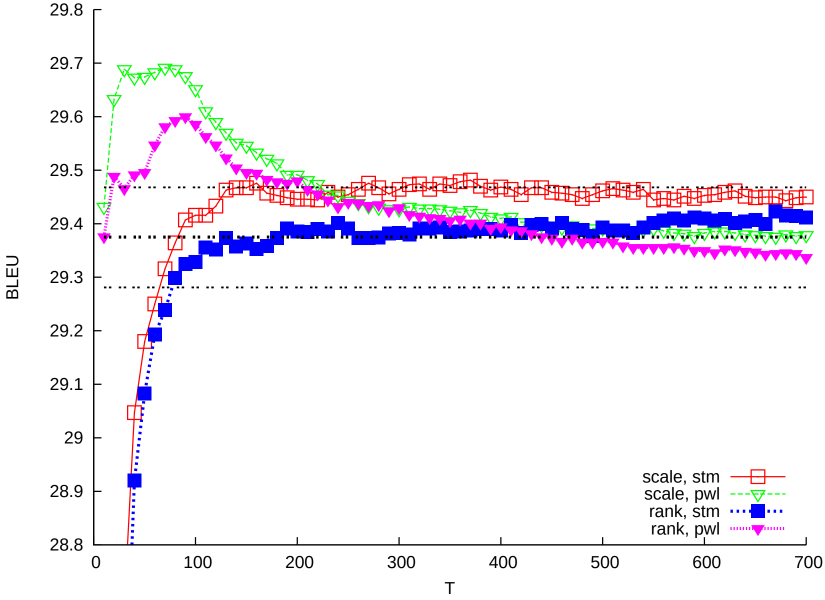
For each feature, we considered:

- the normalized feature value: feature value divided by the number of words *and* phrases
- the scaled feature value: re-scale all features to $[0, 1]$
- the corresponding rank-features: sort according to feature & take its rank
- score of the linear model

configuration	**feature sets**	**#features**
basic	—	12
extended	—	24
basic	scale	33
	scale & rank	45
extended	scale	69
	scale & rank	93

Impact of hyper-parameters

On WMT'10 test set:



Results

Using a validation set:

val./test	WMT'10	WMT'11	WMT'12	MERT	300-best oracle
WMT'10	—	29.68	29.58	29.38	39.72
WMT'11	30.42	—	30.41	30.16	41.11
WMT'12	30.50	30.52	—	30.38	40.64

extended condition, all scores are averaged over 8 runs

- always improving baseline
- still far from oracle scores
- better improvements if using an homogeneous validation set (eg. cross-validation)

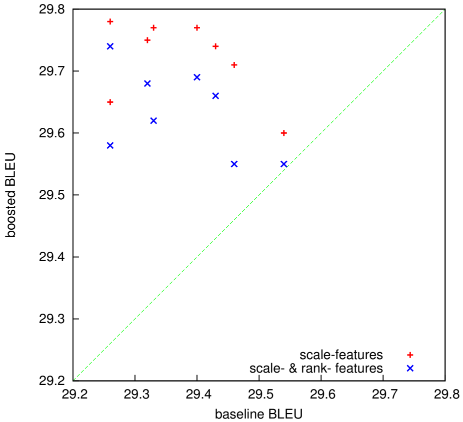
Impact of Non-Linearity

Selection phases of models/features:

1. $T \lesssim 10$ select MERT linear model score
2. $10 \lesssim T \lesssim 50$ use other features, only linear models
3. $50 \lesssim T$ non-linearity starts to appear
4. $T \gtrsim M$: over-fitting

Maximum Relative Gains

Maximum relative gains in BLEU for 8 re-runs on WMT'10:



- worse MERT runs improve more (not surprising)
- reranked worst MERT surpasses best MERT (surprising)

Conclusions

Conclusions

- non-linear approach to reranking n-best lists
- proof-of-concept to avoid tight decoder integration
- approach boosts performance by at least $\mathbf{+0.4}$ BLEU-points

Limits/Future Works

- very small gain $\Rightarrow$ hypotheses translation are selected with a linear function
- future directions:
  - non-linear lattice rescoring / decoding with a non-linear scoring function
  - add more features

Thank you for your attention!

David Chiang, Yuval Marton & Philip Resnik.
Online Large-Margin Training of Syntactic and Structural Translation Features.
In Proc. of EMNLP, pages 224–233, Honolulu, Hawaii, 2008.

David Chiang, Kevin Knight & Wei Wang.
11,001 New Features for Statistical Machine Translation.
In Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 218–226, Boulder, Colorado, June 2009. Association for Computational Linguistics.

Kevin Duh & Katrin Kirchhoff.
Beyond log-linear models: Boosted minimum error rate training for nbest re-ranking.
In Proc. of ACL, Short Papers, 2008.

Michel Galley & Chris Quirk.
Optimal Search for Minimum Error Rate Training.
In Proc. of EMNLP, pages 38–49, July 2011.

Mark Hopkins & Jonathan May.
Tuning as Ranking.
In Proc. of EMNLP, pages 1352–1362, July 2011.

Shankar Kumar, Wolfgang Macherey, Chris Dyer & Franz Och.
Efficient Minimum Error Rate Training and Minimum Bayes-Risk decoding for translation hypergraphs and lattices.
In Proc. of ACL and the Int. Conf. on NLP of the AFNLP, pages 163–171, 2009.

Wolfgang Macherey, Franz Josef Och, Ignacio Thayer & Jakob Uszkoreit.
Lattice-based minimum error rate training for statistical machine translation.
In Proc. of the Conf. on EMNLP, pages 725–734, 2008.

Artem Sokolov & Franois Yvon.
Minimum Error Rate Semi-Ring.
In Mikel Forcada & Heidi Depraetere, editeurs, Proc. of European Conf. on Machine Translation, pages 241–248, Leuven, Belgium, 2011.