Maximum Rank Correlation Training for Statistical Machine Translation

Daqi Zheng*, Yifan He**, Yang Liu*, Qun Liu* **
*ICT, Chinese Academy of Sciences
**CNGL, Dublin City University
Originally published at MT Summit XIII 2011

Presented by Qun Liu
MONOMT Workshop
Attached with AMTA, San Diego US, 1st Nov 2012

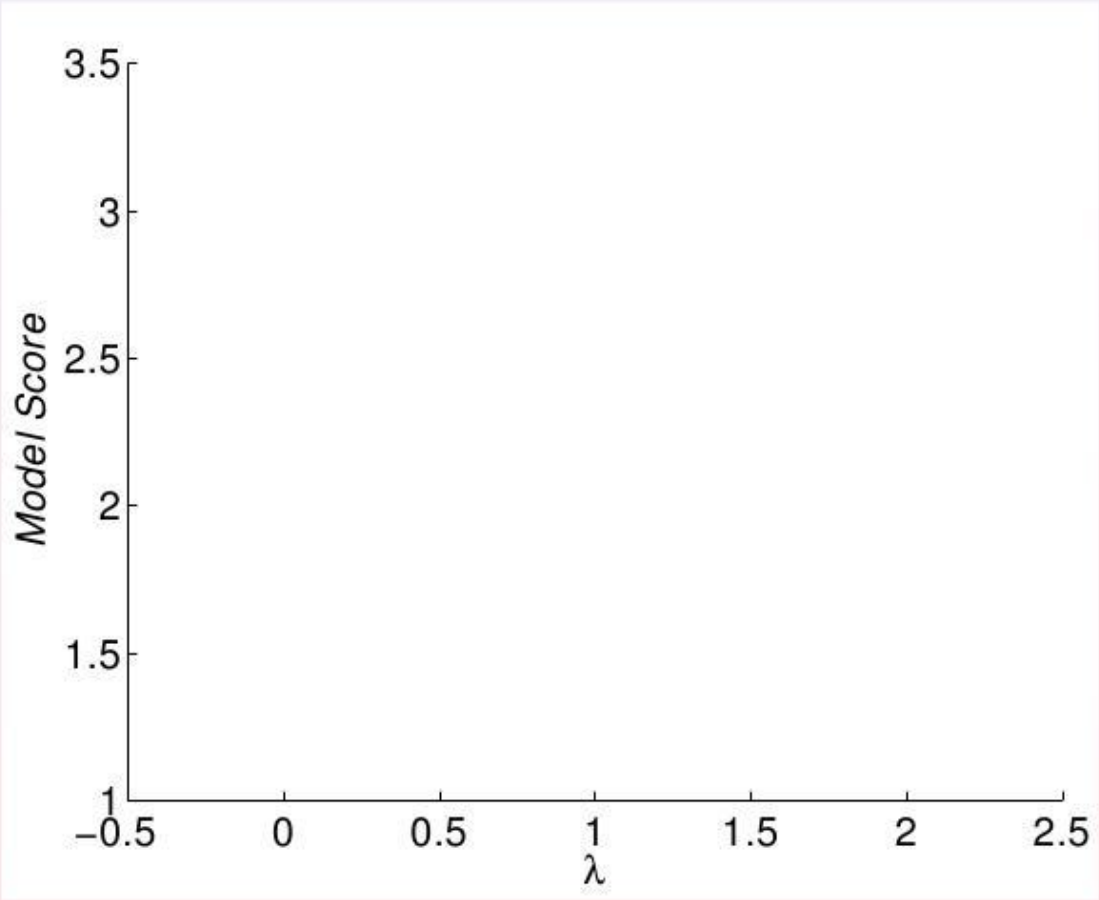
Outline

- Introduction
- Motivation
- Maximum Rank Correlation Training
- Experiments and Results
- Result Analysis
- Conclusion

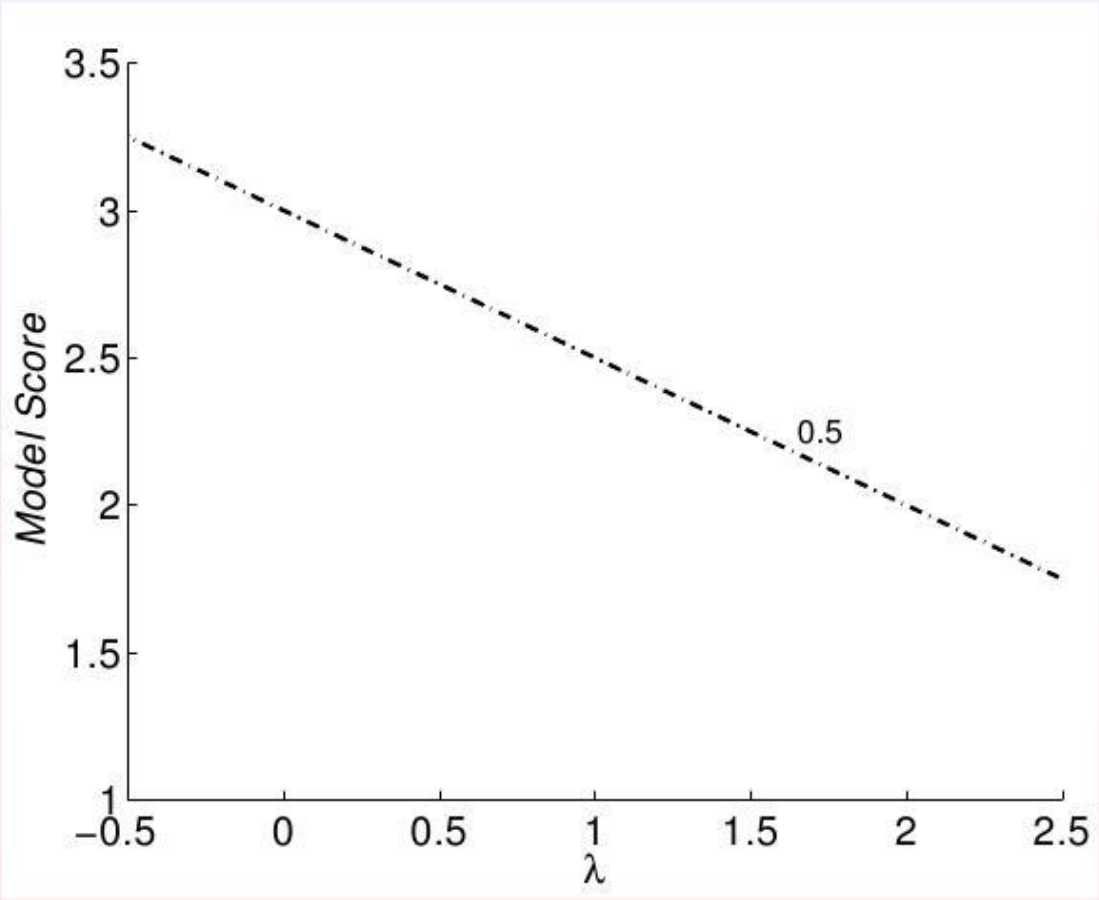
Introduction

- MERT: Minimal Error Rate Training
  - N-best candidates are given by the decoder for each sentences
  - Tune the parameter λ to make the best candidate (with highest BLEU score) to have the highest model score

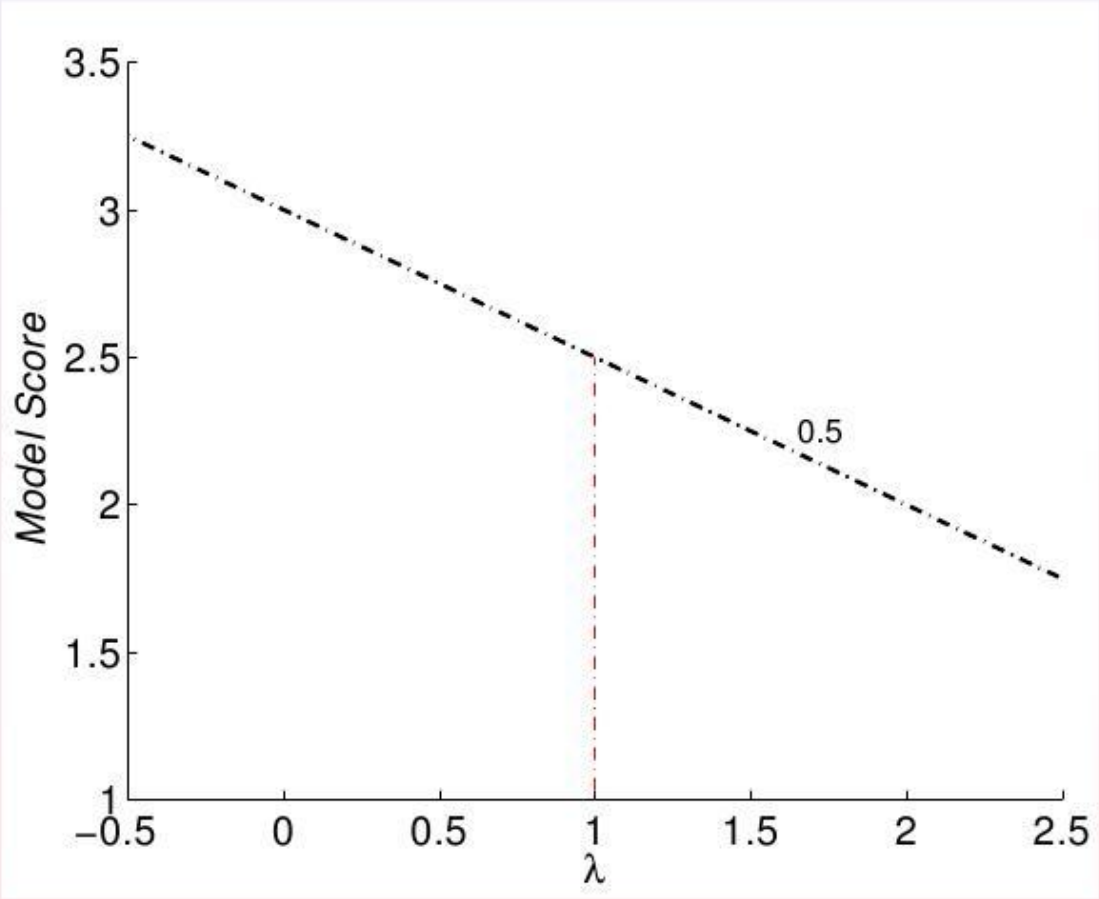
Example



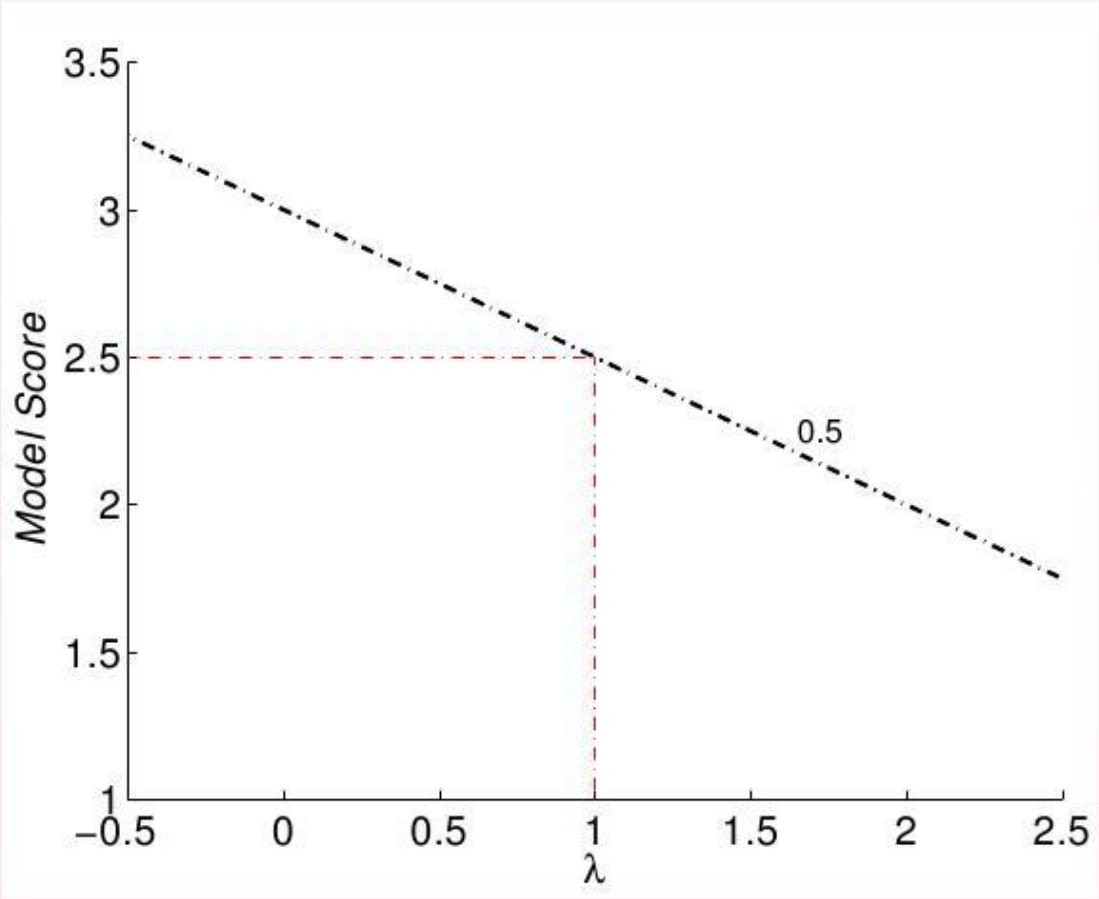
Example



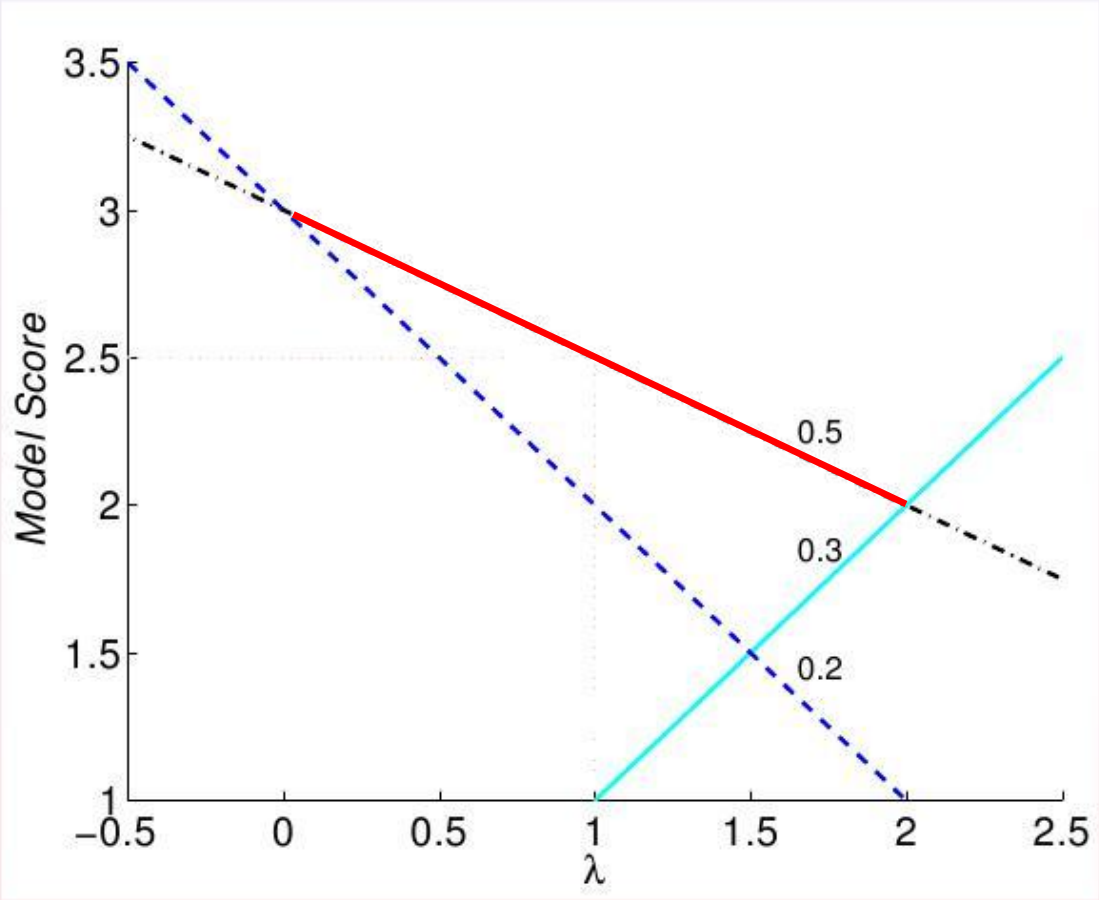
Example



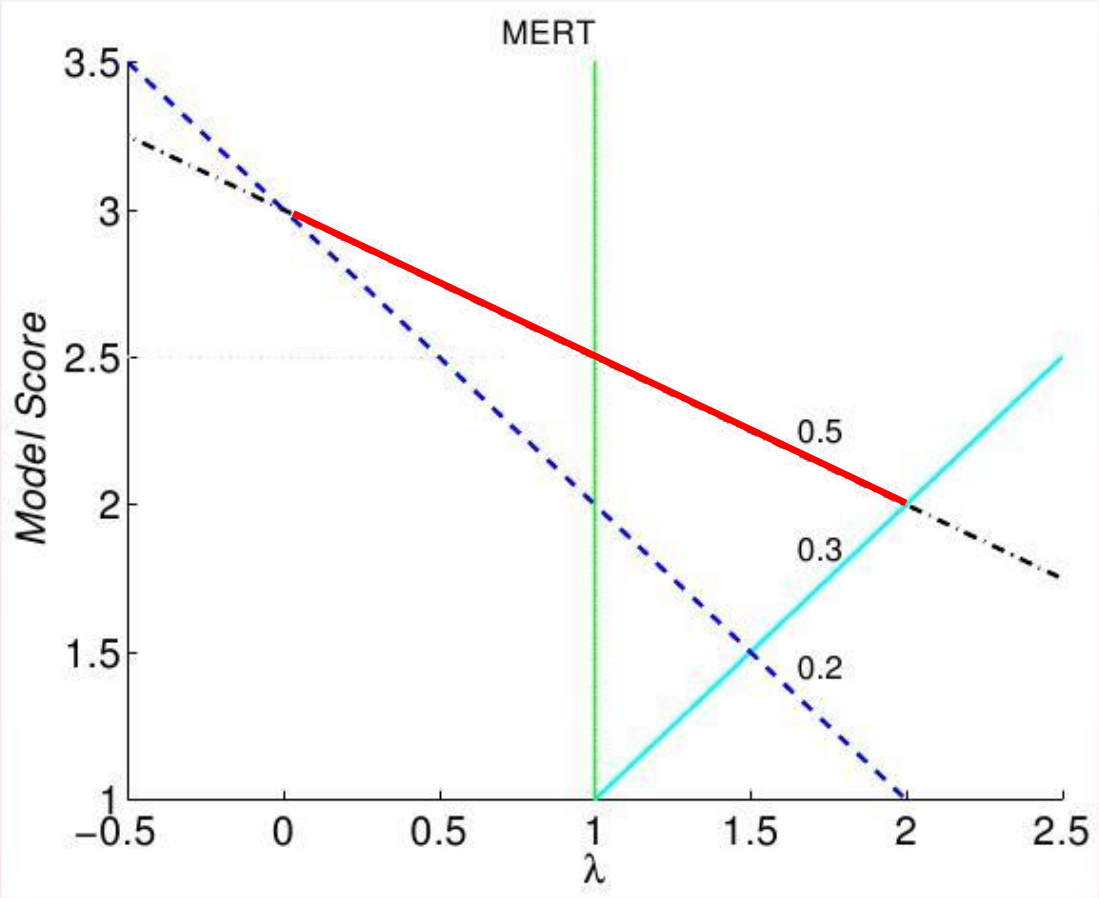
Example



Example



Example



MERT

- Not good for rich features (>20)
- Not stable for local extremums
- Not generalizable across domains

Alternative solution: Min-Risk

[Li & Eisner EMNLP2009]

- Define the Risk as:

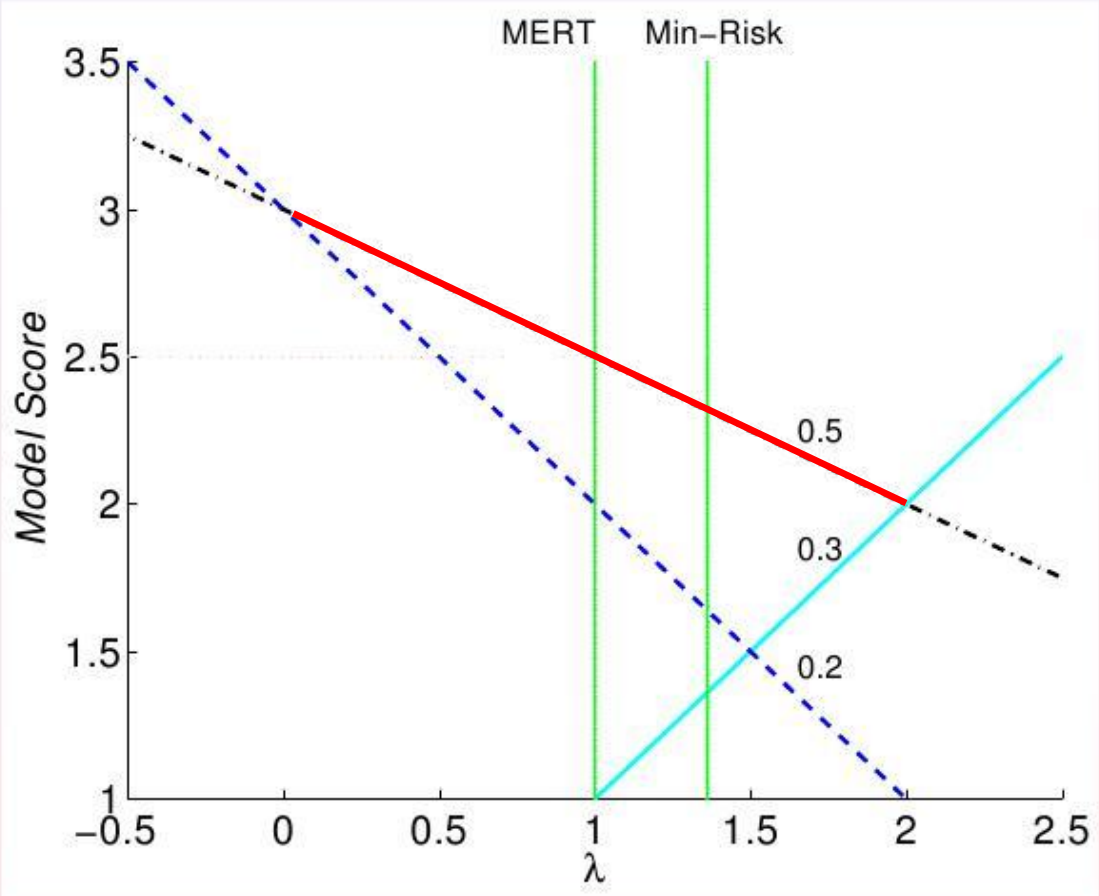
$$R = -\sum_i p(c_i)\,\mathrm{SBLEU}(c_i)$$

  while the Posterior Probability can be defined using model score, for example:

$$p(c_i) = \exp(\gamma \cdot score(c)) / \sum_i \exp(\gamma \cdot score(c_i))$$

- Tune the parameter λ to minimize the Risk

Example

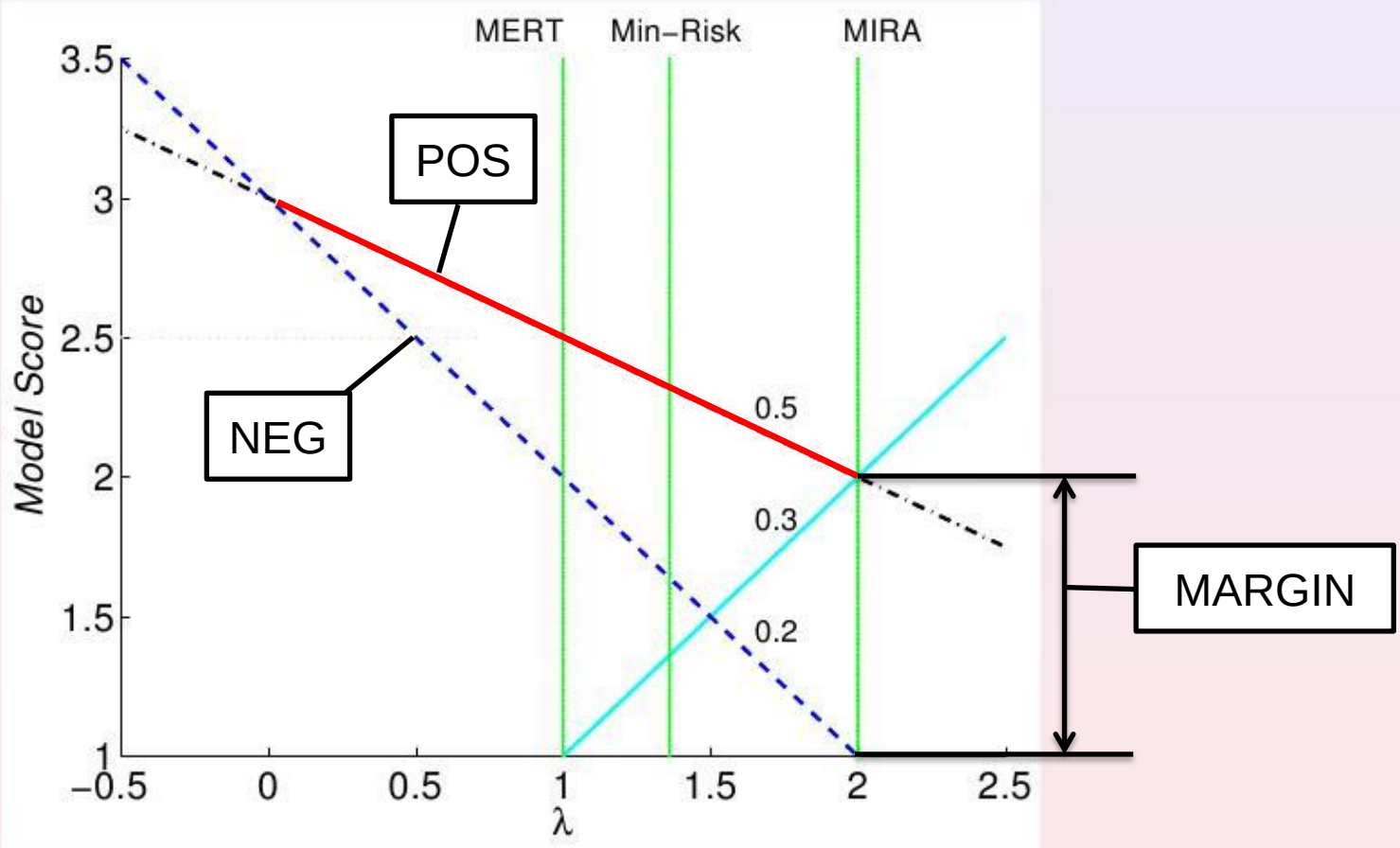


Alternative solution: MIRA

[Chiang et al. EMNLP2008]

- Select a Positive Set of candidates with high BLEU scores
- Select a Negative Set of candidates with low BLEU score
- Tune the parameter λ to maximize the difference (margin) of the model score between Negative Set and that of Positive Set.

Example


MERT
Min–Risk
MIRA
POS
NEG
Model Score
3.5
3
2.5
2
1.5
1
−0.5
0
0.5
1
1.5
2
2.5
λ
0.5
0.3
0.2
MARGIN

Outline

Introduction

Motivation

Maximum Rank Correlation Training

Experiments and Results

Result Analysis

Conclusion

Our Motivation: Max Rank Correlation

- We would like to choose the λ which maximize the correlation between the ranking of the candidates according the model scores and that according to the BLEU scores

Motivation

- For example: 8 candidates
  - BLEU score ranking: 1 2 3 4 5 6 7 8
  - Model score ranking with λ_1: 1 8 7 6 5 4 3 2
  - Model score ranking with λ_2: 2 1 3 4 5 6 7 8
- For MERT, λ_1 will be chosen
- We would like to choose λ_2

Outline

Introduction

Motivation

Maximum Rank Correlation Training

Experiments and Results

Result Analysis

Conclusion

Maximum Rank Correlation (MRC)

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \left(\sum_{i=1}^{M} w_i \cdot Corr_i(\boldsymbol{\lambda}) \right)$$

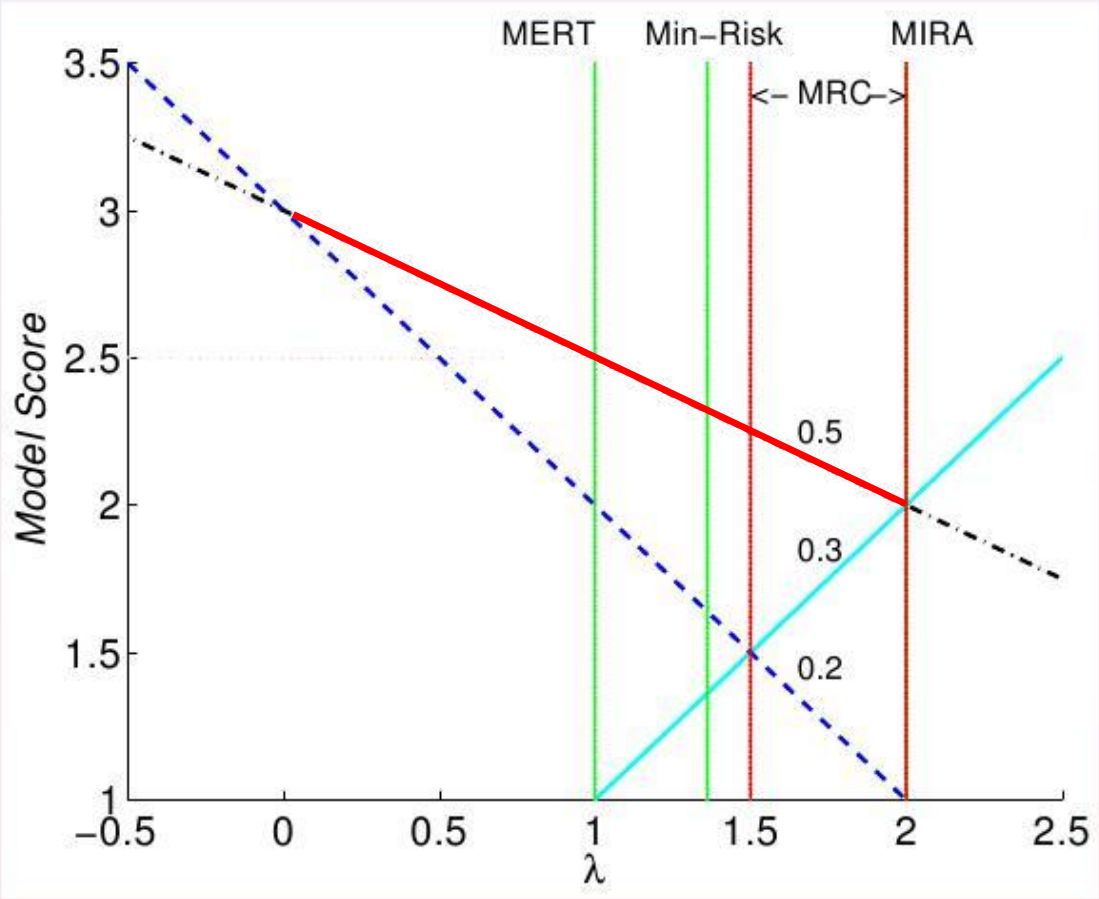
$$Corr_i(\boldsymbol{\lambda}) = Corr(\Phi_1^N(\boldsymbol{\lambda}), SBLEU(\mathbf{e}_1^N)))$$

Spearman Rank Correlation Coefficient →

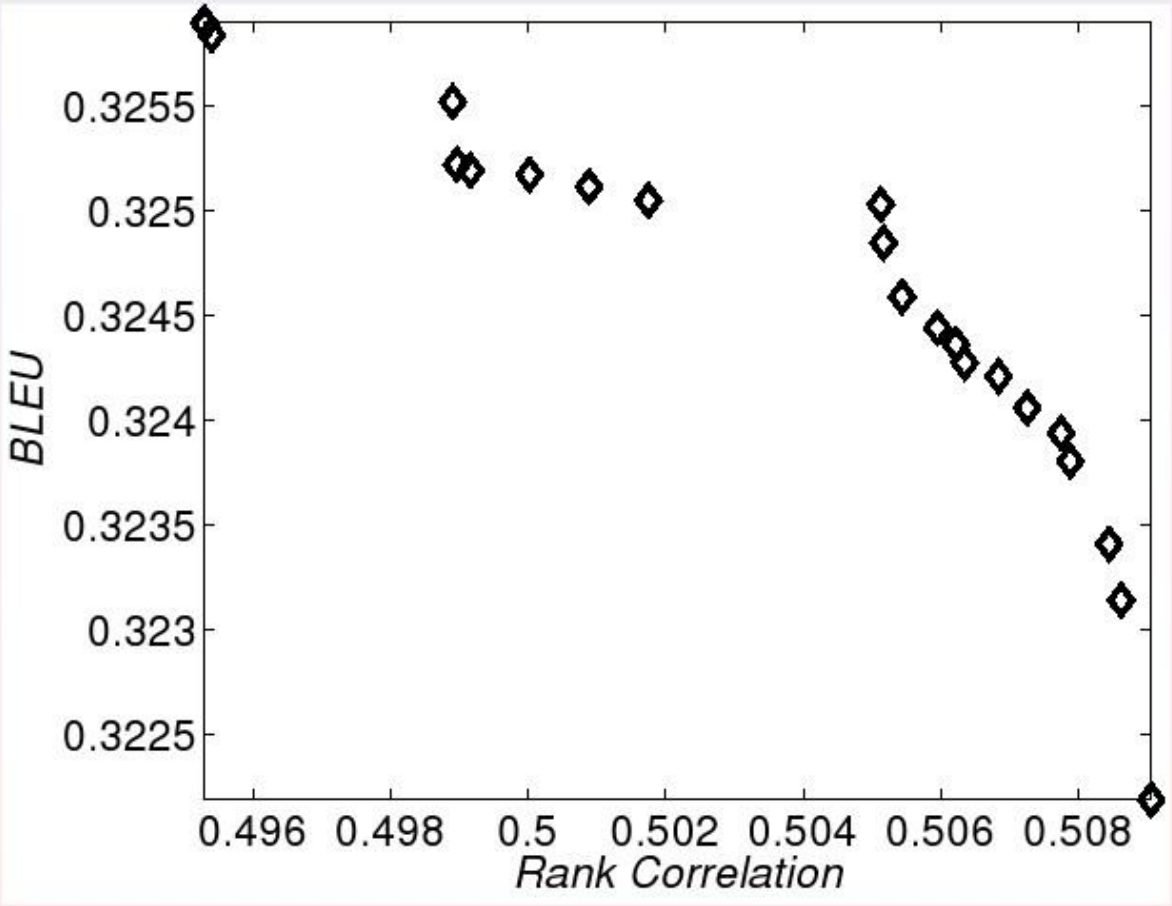
$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$$

$$w_i = length(\mathbf{f}_i) / \sum_{i=1}^{M} (length(\mathbf{f}_i))$$

Example


MERT
Min−Risk
MIRA
<− MRC−>
3.5
3
2.5
2
1.5
1
Model Score
0.5
0.3
0.2
−0.5
0
0.5
1
1.5
2
2.5
λ

Rank Correlation



Combination of MER and MRC

$$\hat{\lambda} = \arg\max_{\lambda}(\alpha \cdot \sum_{i=1}^{M} Corr_i(\lambda) + (1-\alpha) \cdot BLEU(\lambda))$$

Multi-objective Optimization

- We use multi-objective evolutionary algorithm (MOEA) (Fonseca et al., 1993) for training.
- We choose an effective MOEA tool: NSGA-II in our experiments.

Outline

Introduction

Motivation

Maximum Rank Correlation Training

Experiments and Results

Result Analysis

Conclusion

Experiment Settings

- Data
  - French-English WMT08 shared translation task
  - Training data : Europarl v3b release
  - Language model : English part of monolingual language model
  - training data
  - Tuning set: dev2006
  - In-domain test sets : test2006, test2007, test2008
  - Out-of-domain test sets : newstest2008, newstest2009, newstest2010
- System
  - Machine Translation: Moses Suite
  - Spearman Rank Correlation Coefficient: goose
  - Multi-Objective Optimization: NSGA-II

Generic Algorithm Settings

- First Generation
  - 10 individuals from MERT training
  - 390 individuals randomly generated
- Evolution
  - 100 generations
  - 400 individuals for each generation

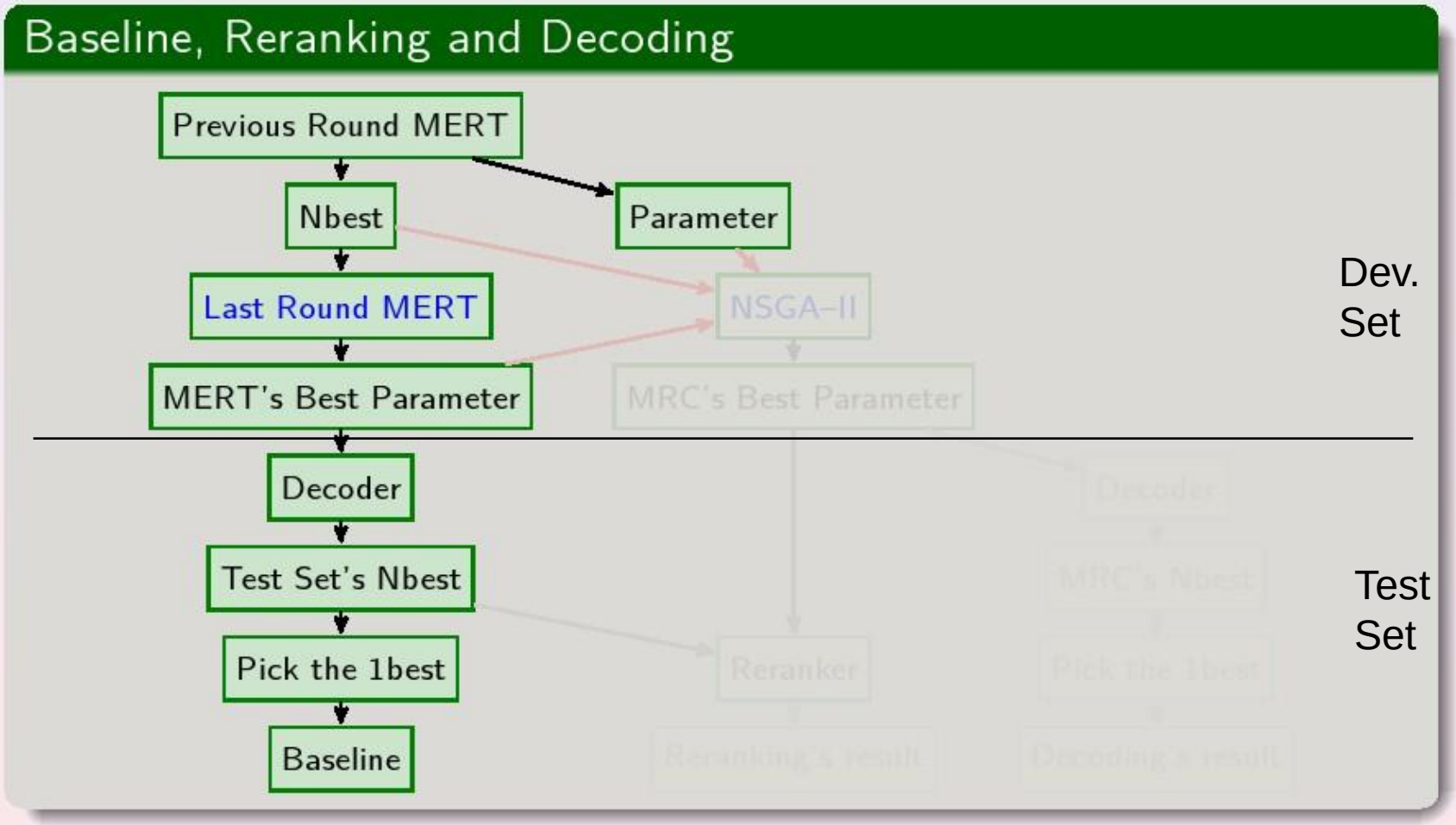
Experiment Process

Baseline

Reranking

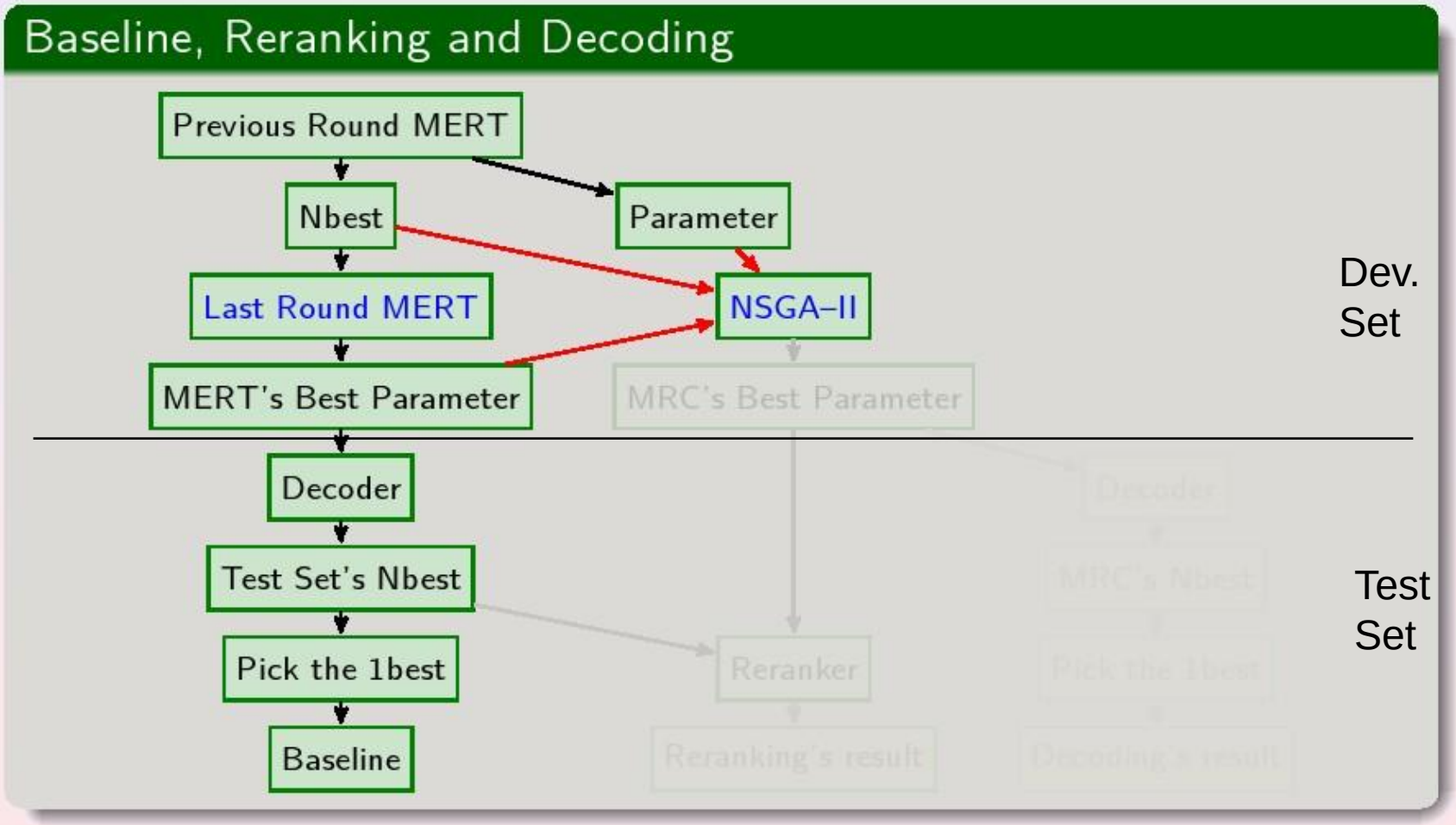
Decoding

Process

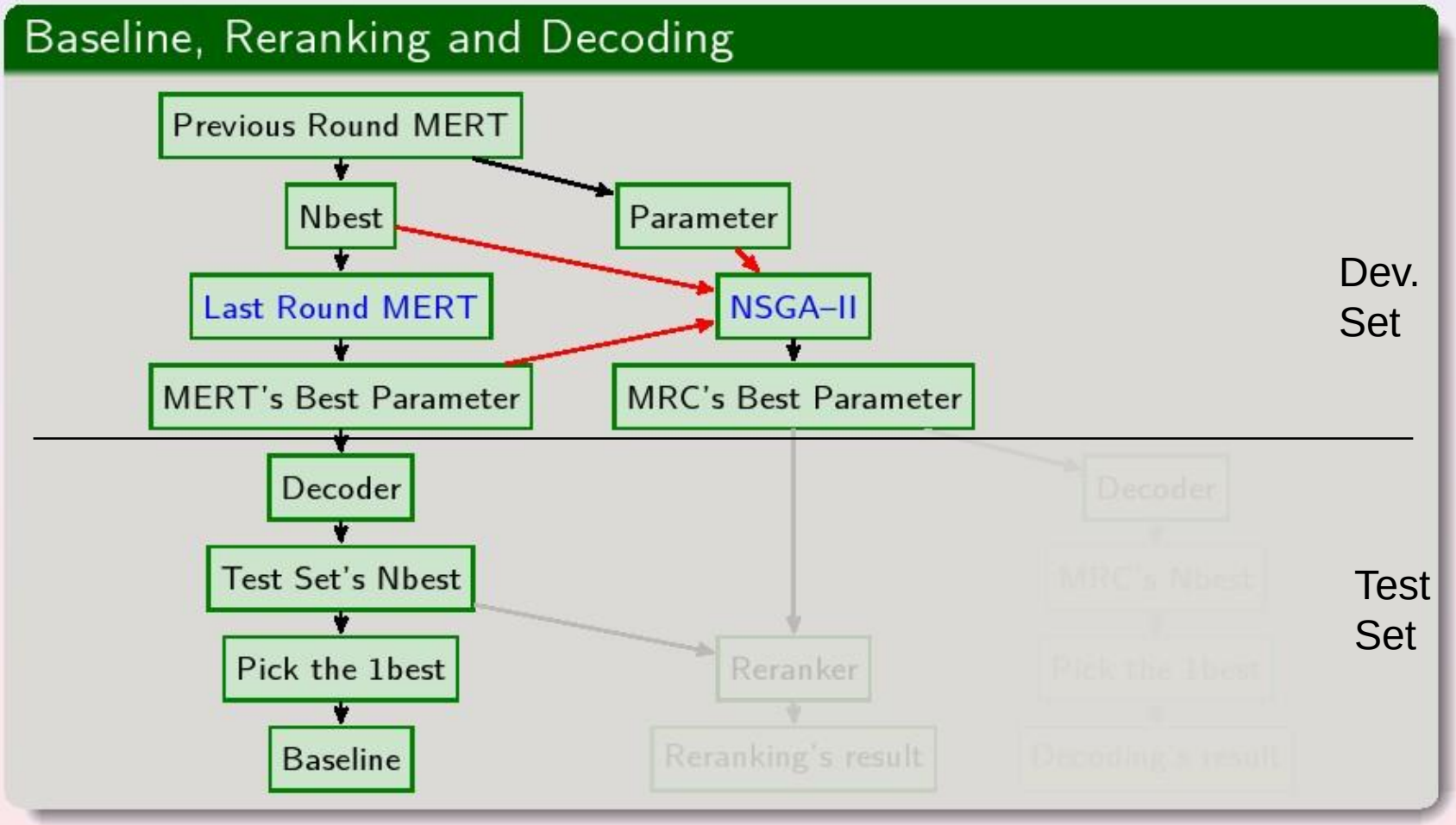


Baseline

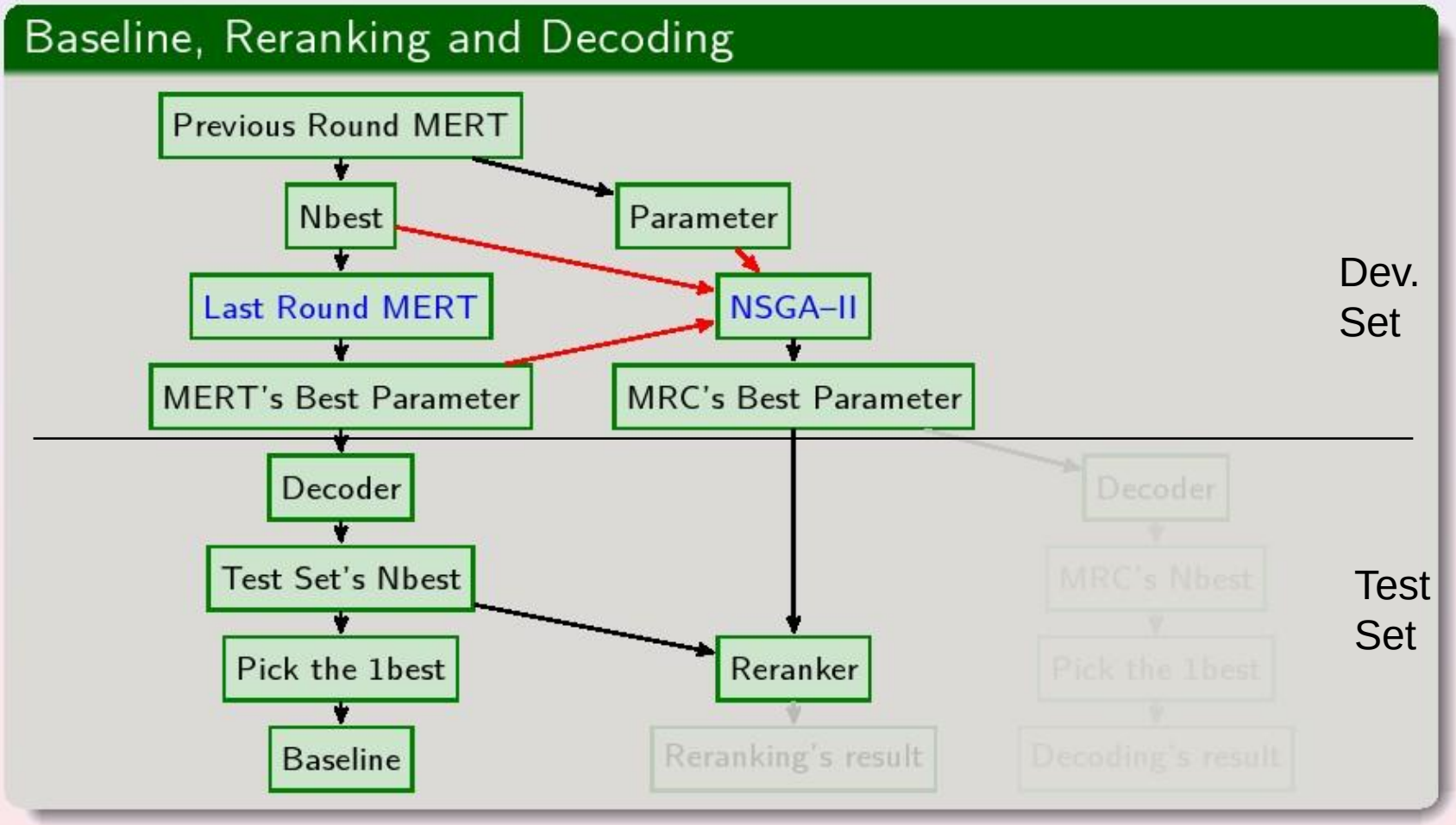
Process


Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Test Set's Nbest
Pick the 1best
Baseline
Reranker
Reranking's result
Decoding's result
Test Set
Baseline
Reranking

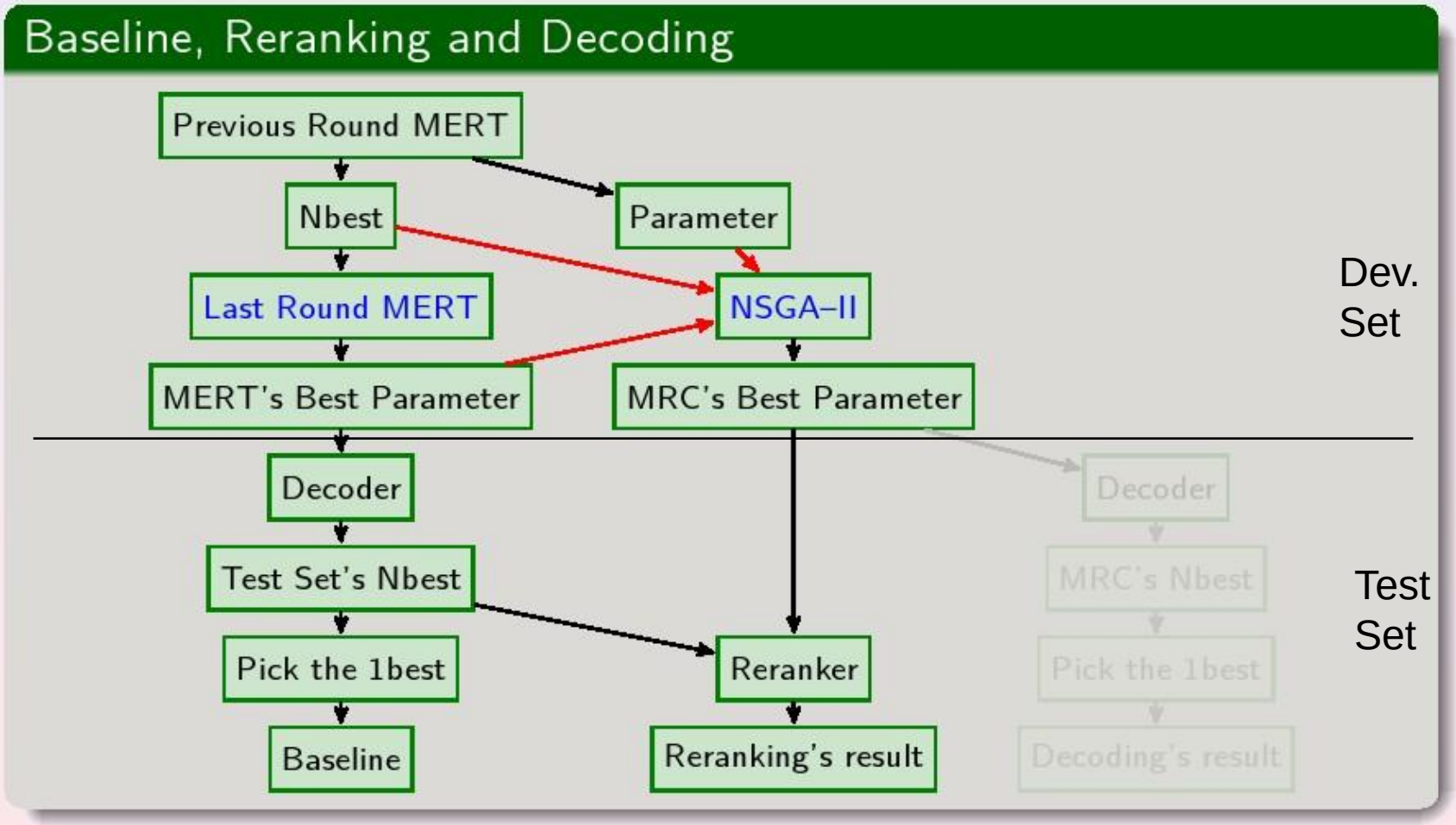
Process


Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Test Set's Nbest
Pick the 1best
Baseline
Reranker
Reranking's result
Test Set
Baseline
Reranking

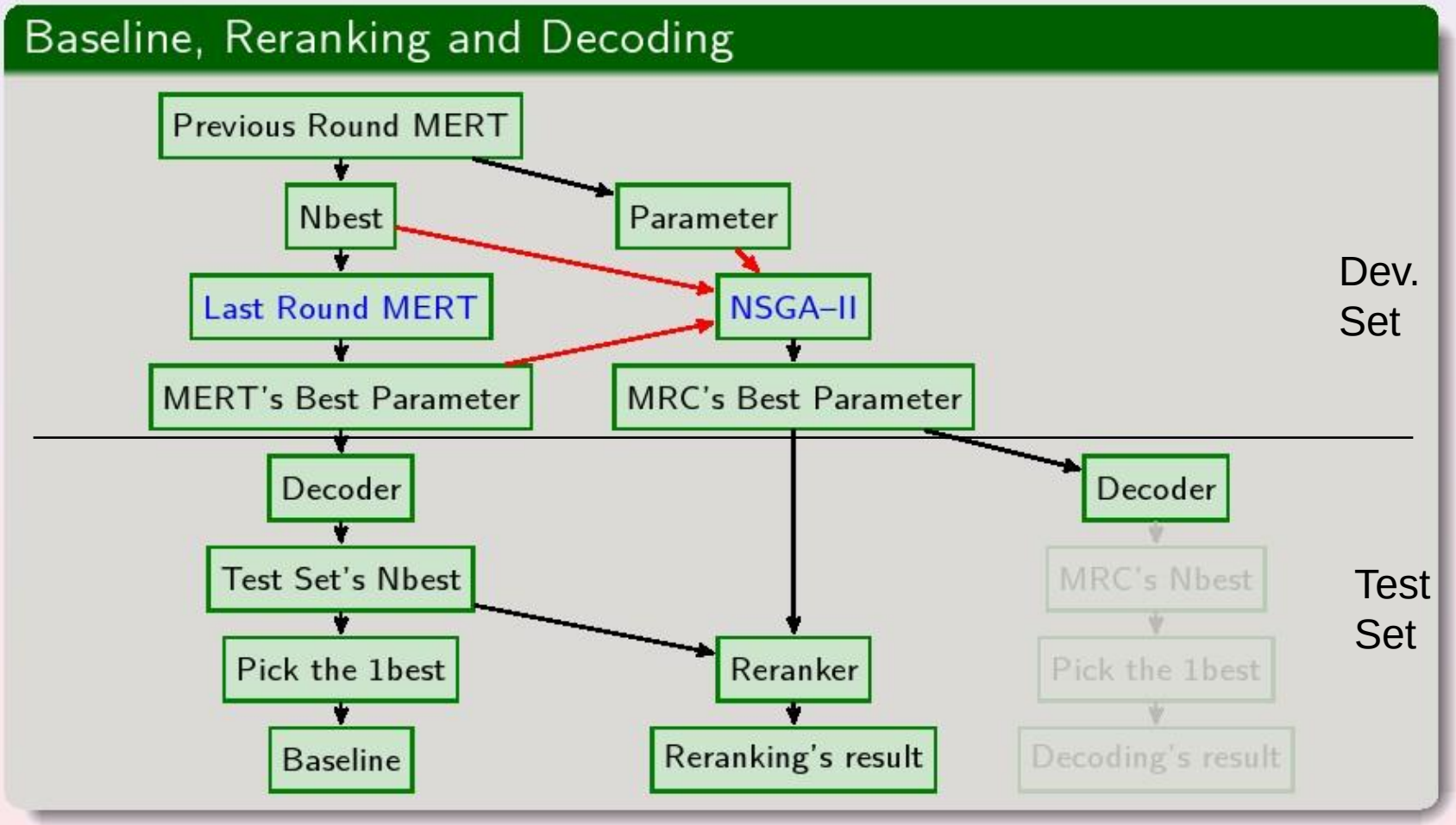
Process


Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Test Set's Nbest
Pick the 1best
Baseline
Reranker
Reranking's result
Decoder
MRC's Nbest
Pick the 1best
Decoding's result
Test Set
Baseline
Reranking

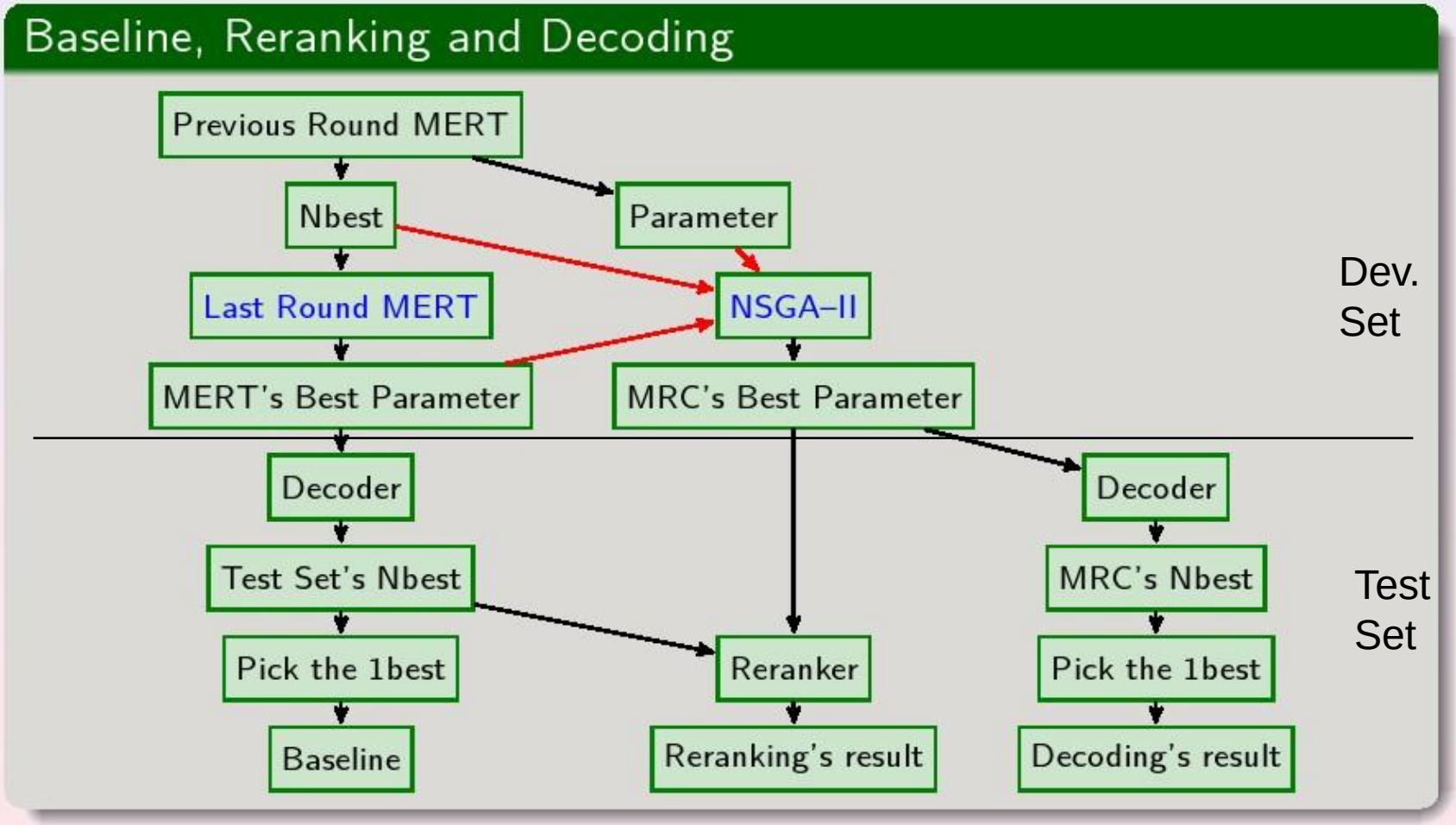
Process


Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Test Set's Nbest
Pick the 1best
Baseline
Reranker
Reranking's result
Decoder
MRC's Nbest
Pick the 1best
Decoding's result
Test Set
Baseline
Reranking

Process


Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Decoder
Test Set's Nbest
MRC's Nbest
Test Set
Pick the 1best
Reranker
Pick the 1best
Baseline
Reranking's result
Decoding's result
Baseline
Reranking
Decoding

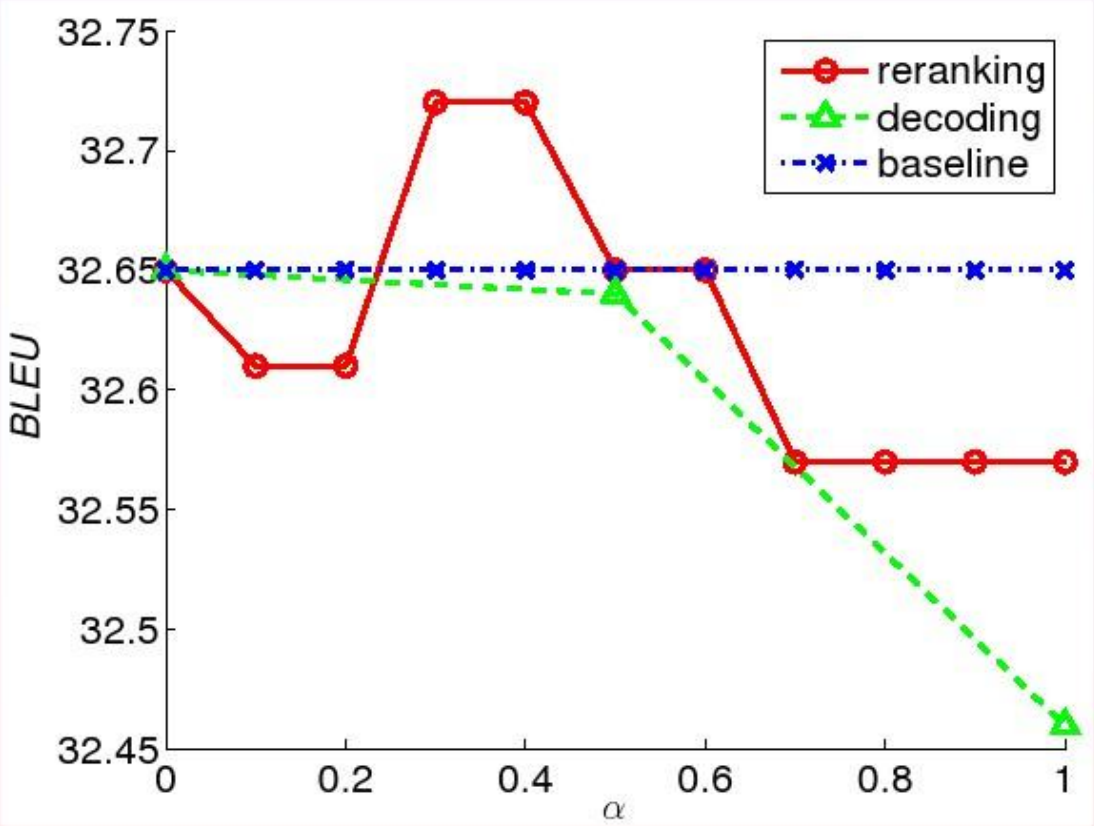
Process


Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Test Set's Nbest
Pick the 1best
Baseline
Reranker
Reranking's result
Decoder
MRC's Nbest
Pick the 1best
Decoding's result
Test Set
Baseline
Reranking
Decoding

Results

- **Best α on development set**
- Results via different α on test set
- Improvement of reranking on each MERT tuning run
- Improvement of reranking on different genetic algorithm settings
- Time cost

Result of Best α on Dev Set



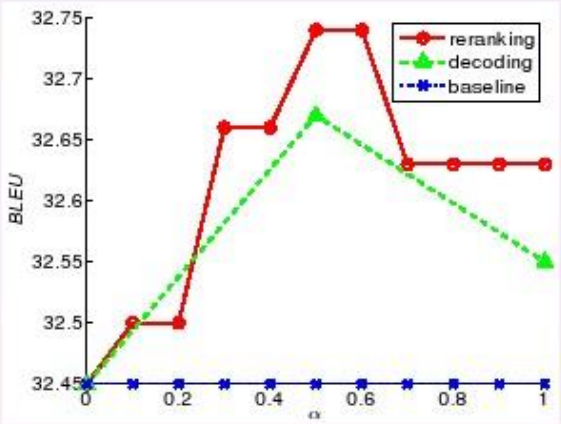
Results

- Best α on development set
- **Results via different α on test set**
- Improvement of reranking on each MERT tuning run
- Improvement of reranking on different genetic algorithm settings
- Time cost

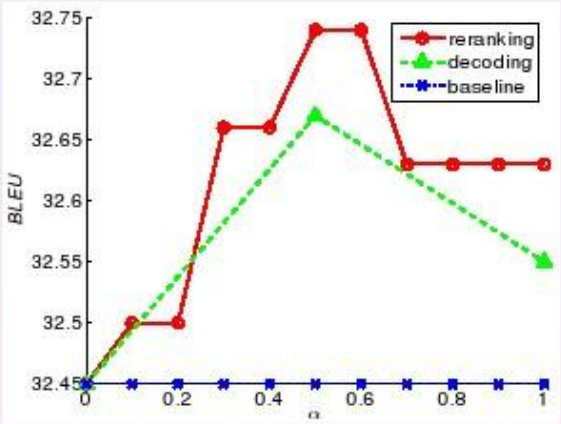
Result via Different α

In-Domain

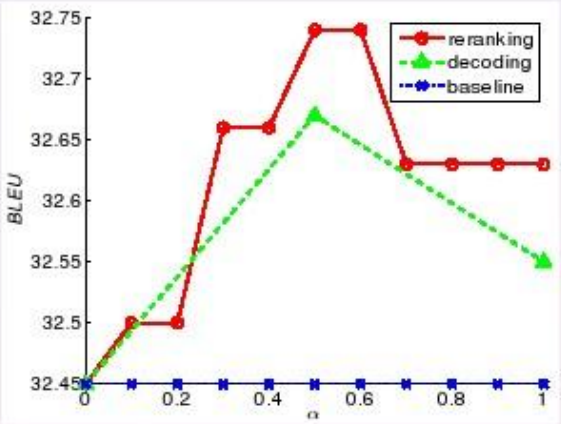
test2006



test2007



test2008

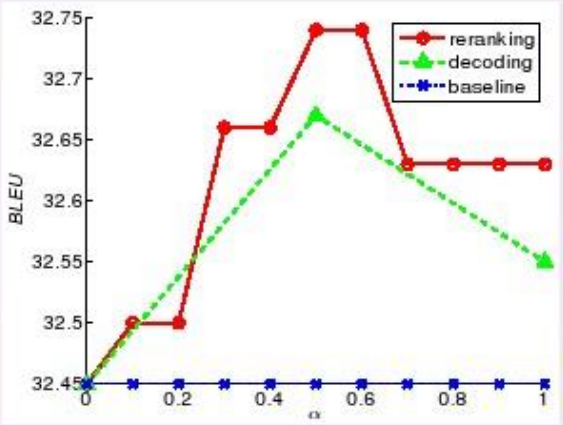


Out-of-Domain

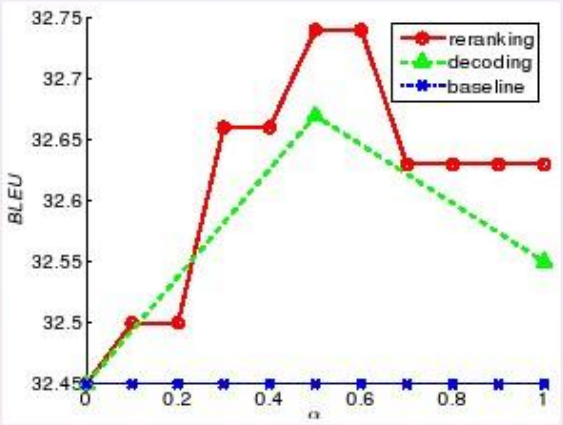
Result via Different α

In-Domain

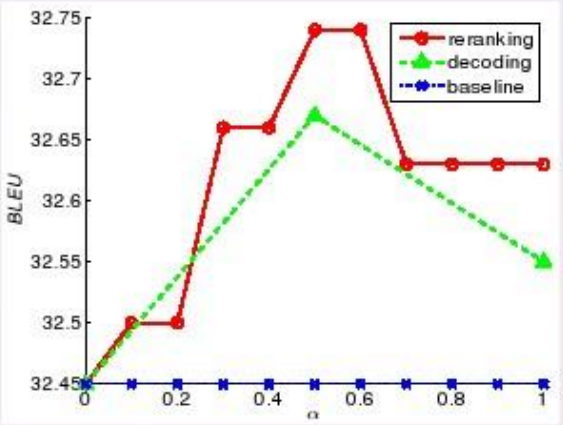
test2006



test2007

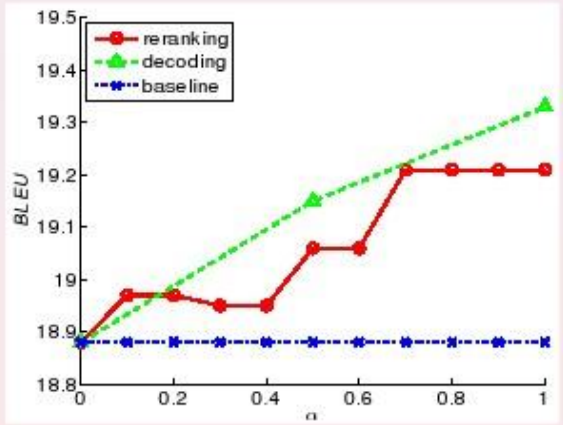


test2008

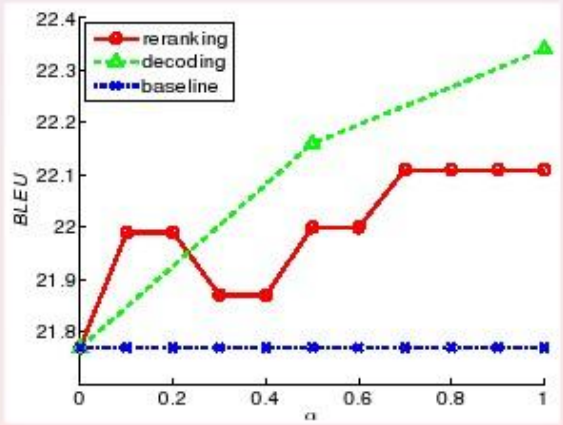


Out-of-Domain

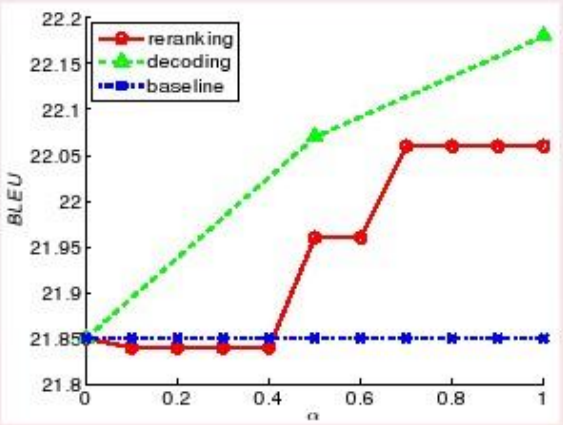
newstest2008



newstest2009



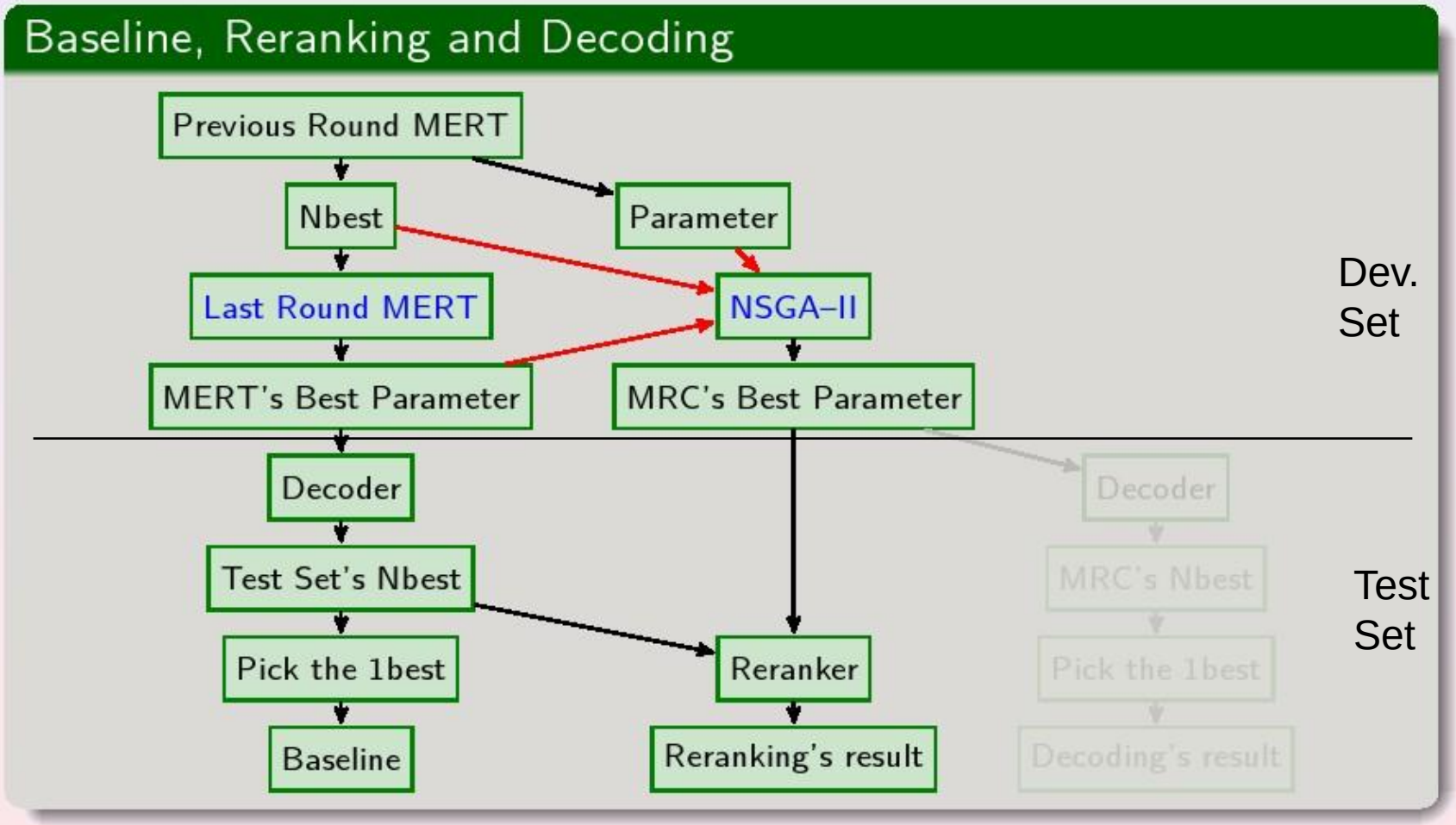
newstest2010



Results

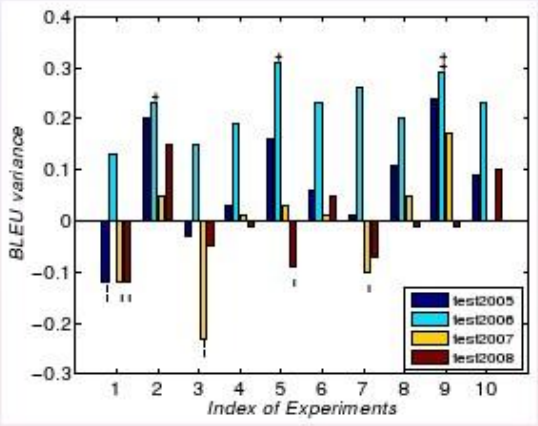
- Best α on development set
- Results via different α on test set
- **Improvement of reranking on each MERT tuning run**
- Improvement of reranking on different genetic algorithm settings
- Time cost

Process

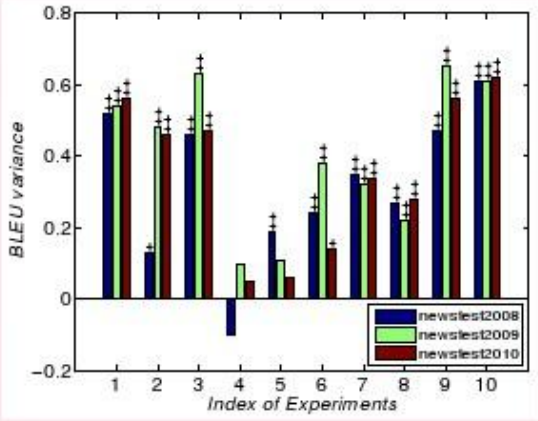

Baseline, Reranking and Decoding
Previous Round MERT
Nbest
Parameter
Last Round MERT
NSGA–II
MERT's Best Parameter
MRC's Best Parameter
Dev. Set
Decoder
Test Set's Nbest
Pick the 1best
Baseline
Reranker
Reranking's result
Decoder
MRC's Nbest
Pick the 1best
Decoding's result
Test Set
Baseline
Reranking

Improvement of Reranking on Each Run

In-Domain

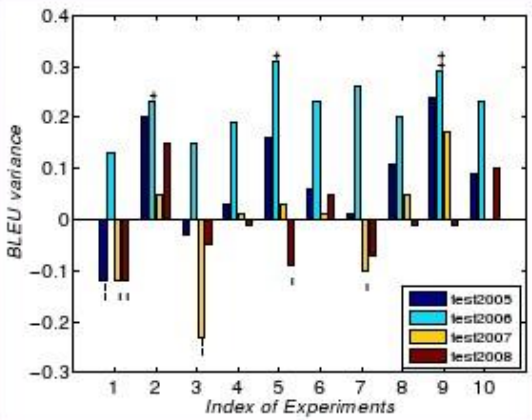


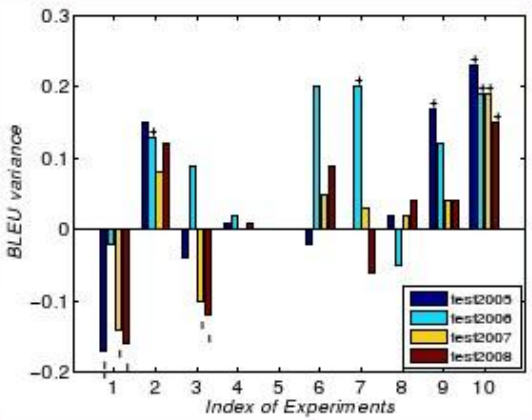
Out-of-Domain



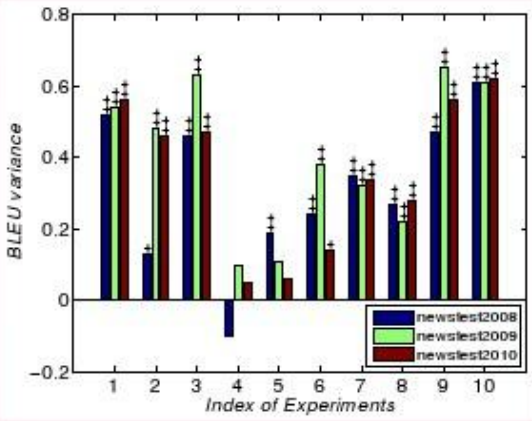
Improvement of Reranking on Each Run

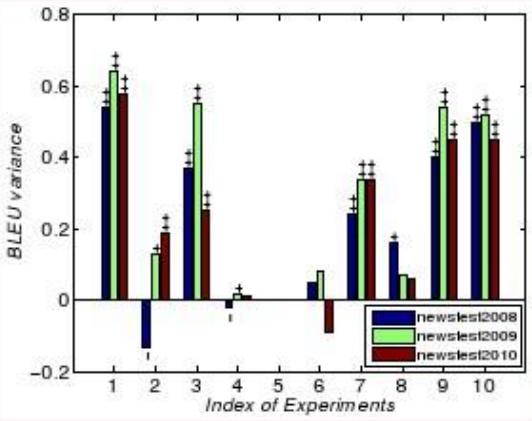
In-Domain





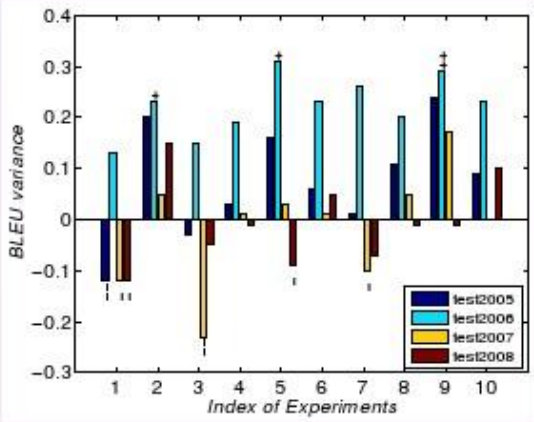
Out-of-Domain

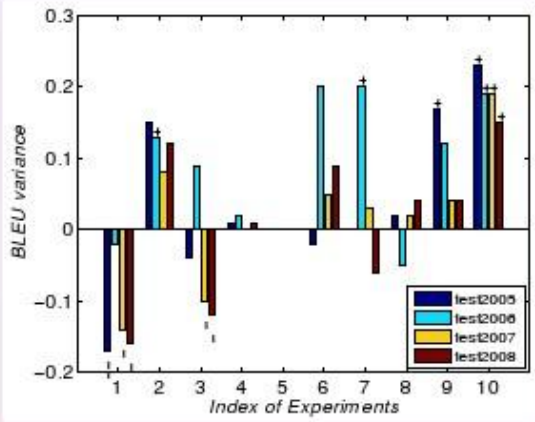


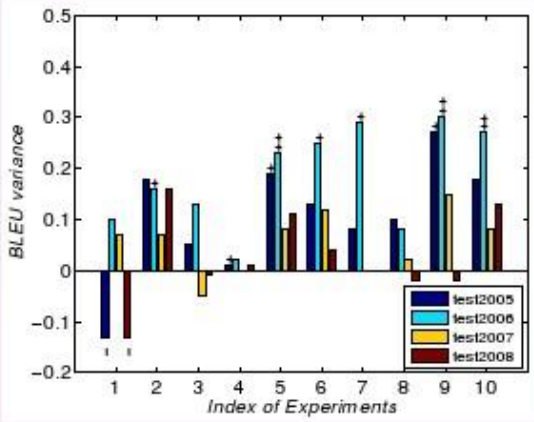


Improvement of Reranking on Each Run

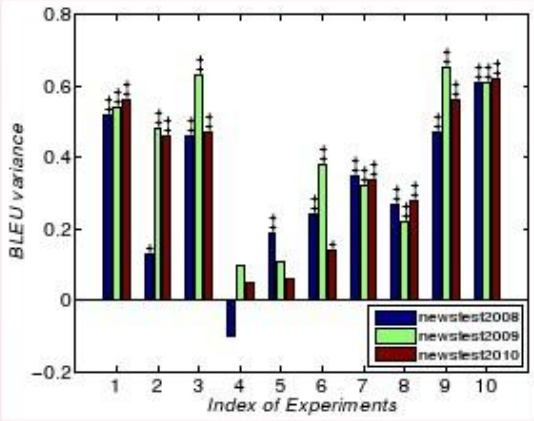
In-Domain

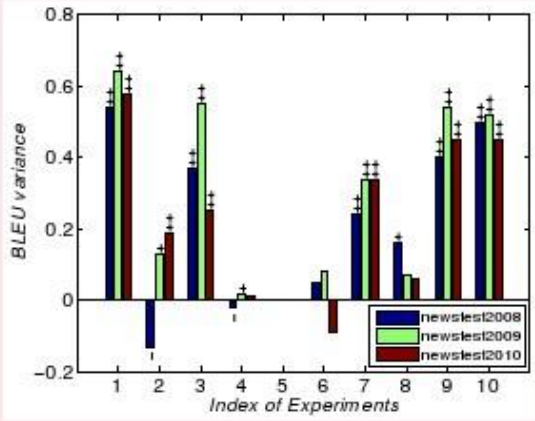


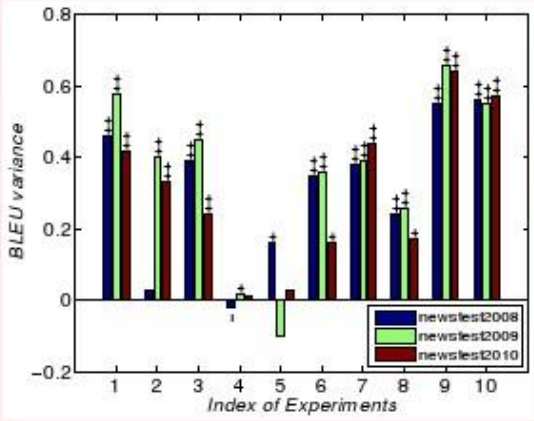




Out-of-Domain



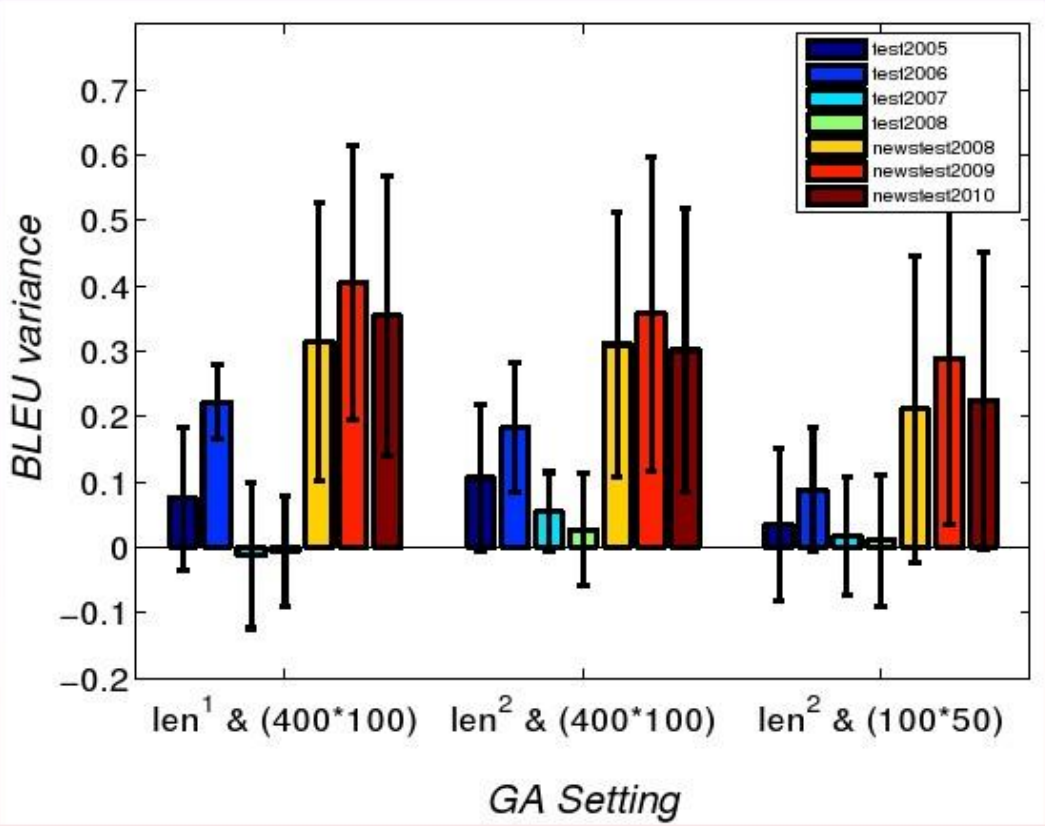




Results

- Best α on development set
- Results via different α on test set
- Improvement of reranking on each MERT tuning run
- **Improvement of reranking on different genetic algorithm settings**
- Time cost

Result Summary



Results

- Best α on development set
- Results via different α on test set
- Improvement of reranking on each MERT tuning run
- Improvement of reranking on different genetic algorithm settings
- **Time cost**

Time Cost

Index of Exps	1	2	3	4	5	6	7	8	9	10
# of Iteration	11	15	15	12	9	10	15	12	14	15
Last Mert	26	22	28	23	17	13	23	12	18	26
Tuning	1222	1766	1657	1329	1047	1113	1694	1363	1613	1741
$len^1\&(400 \cdot 100)$	780	615	782	793	570	448	755	499	606	793
$len^2\&(400 \cdot 100)$	768	624	769	778	569	431	718	499	607	741
$len^2\&(100 \cdot 50)$	95	76	96	97	70	55	89	63	75	96

- 10 experiment's Running Time: in 100 seconds. Compare the GA with the total tuning time, and consider it need only run once at the tuning phase, the computation cost is affordable.

Time Cost

Index of Exps	1	2	3	4	5	6	7	8	9	10
# of Iteration	11	15	15	12	9	10	15	12	14	15
Last Mert	26	22	28	23	17	13	23	12	18	26
Tuning	1222	1766	1657	1329	1047	1113	1694	1363	1613	1741
$len^1\&(400 \cdot 100)$	780	615	782	793	570	448	755	499	606	793
$len^2\&(400 \cdot 100)$	768	624	769	778	569	431	718	499	607	741
$len^2\&(100 \cdot 50)$	95	76	96	97	70	55	89	63	75	96

- 10 experiment's Running Time: in 100 seconds. Compare the GA with the total tuning time, and consider it need only run once at the tuning phase, the computation cost is affordable.

Time Cost

Index of Exps	1	2	3	4	5	6	7	8	9	10
# of Iteration	11	15	15	12	9	10	15	12	14	15
Last Mert	26	22	28	23	17	13	23	12	18	26
Tuning	1222	1766	1657	1329	1047	1113	1694	1363	1613	1741
$len^1\&(400 \cdot 100)$	780	615	782	793	570	448	755	499	606	793
$len^2\&(400 \cdot 100)$	768	624	769	778	569	431	718	499	607	741
$len^2\&(100 \cdot 50)$	95	76	96	97	70	55	89	63	75	96

- 10 experiment's Running Time: in 100 seconds. Compare the GA with the total tuning time, and consider it need only run once at the tuning phase, the computation cost is affordable.

Outline

Introduction

Motivation

Maximum Rank Correlation Training

Experiments and Results

Result Analysis

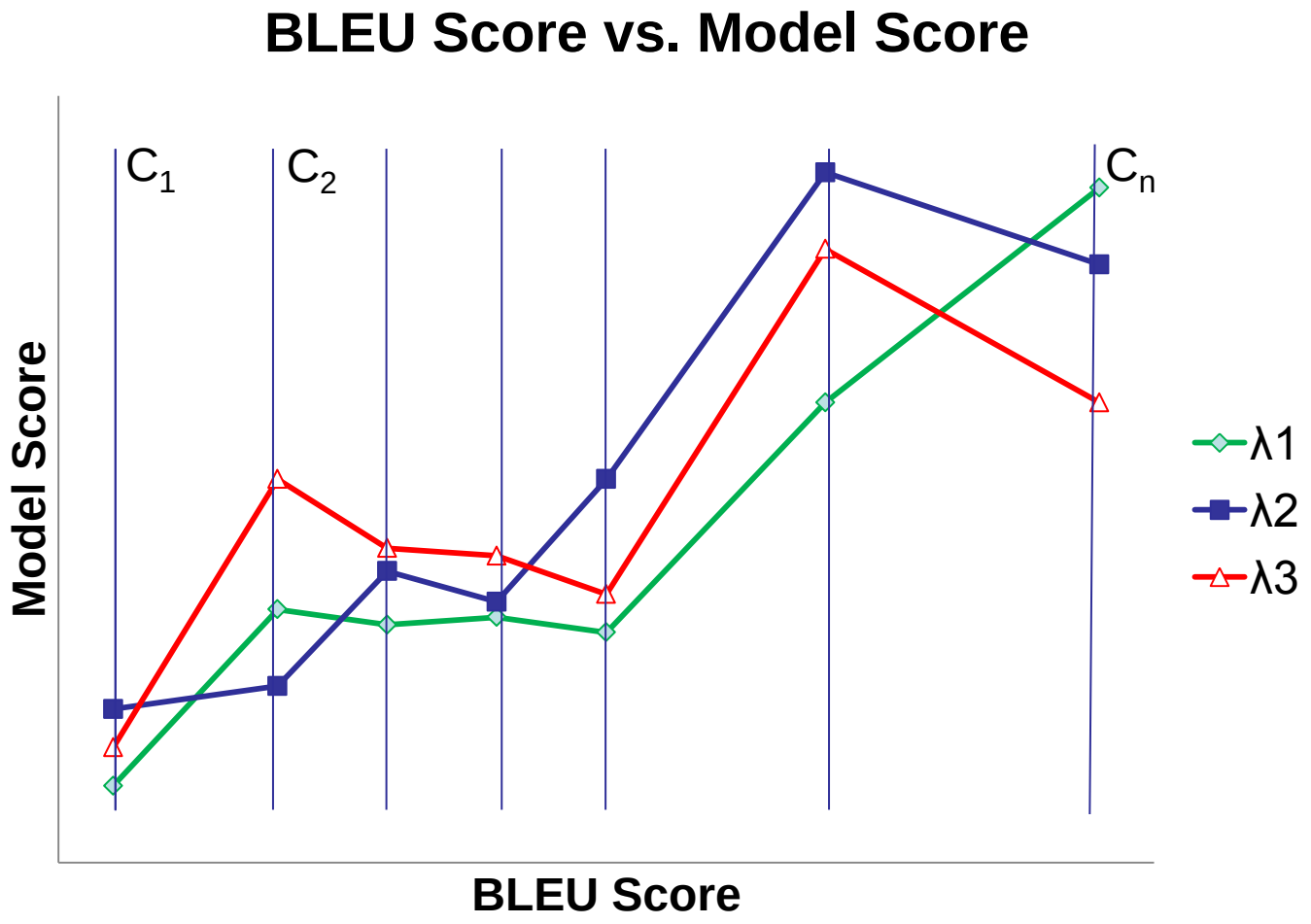
Conclusion

Result Analysis

- MERT+MRCT outperforms MERT both for in-domain and out-of-domain test data

- Why?

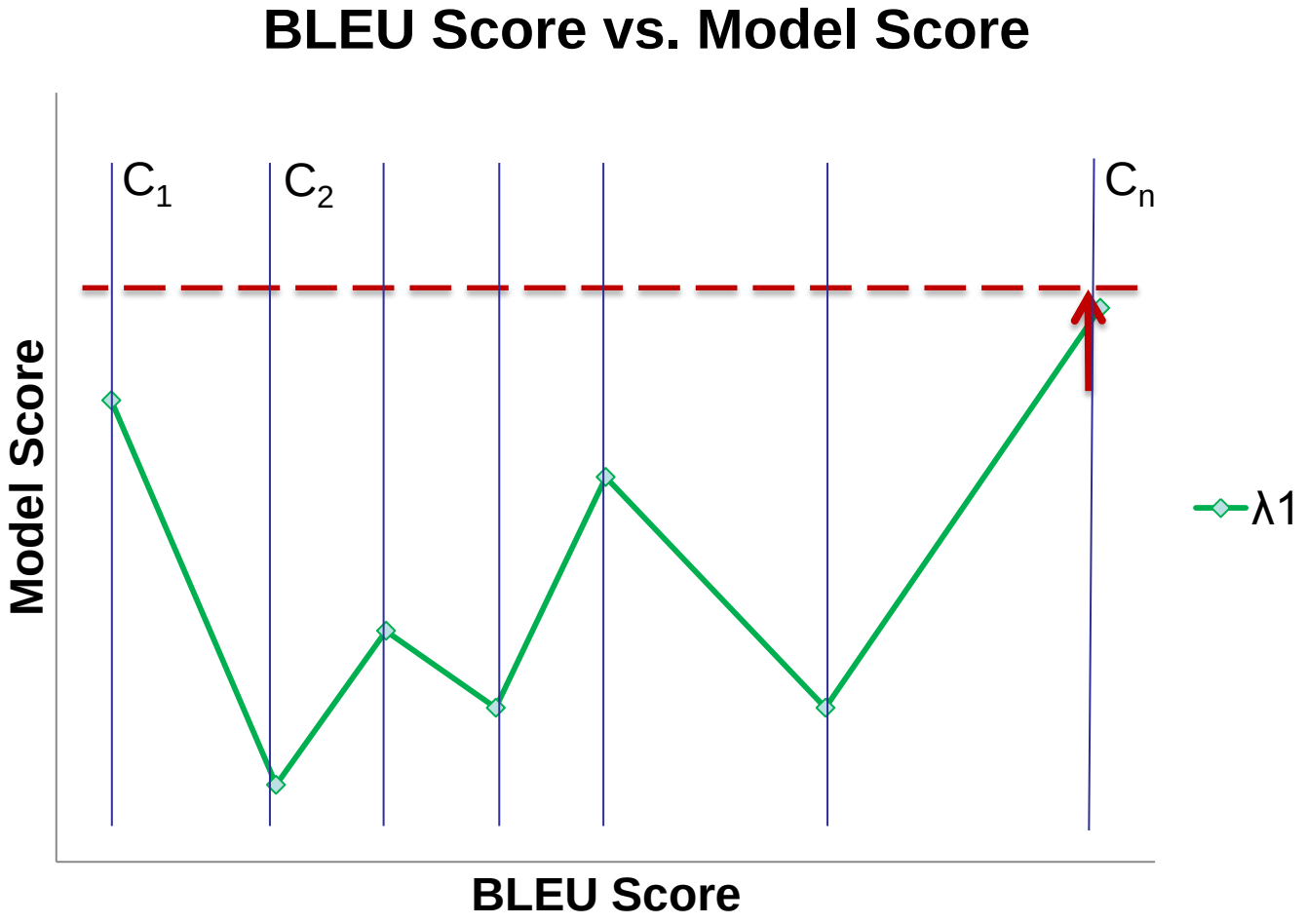
BLEU Score vs. Model Score



MER Training

- MER Training tries to make the right most dot at the highest position
- MER Training does not care if the rest of the line is monotone

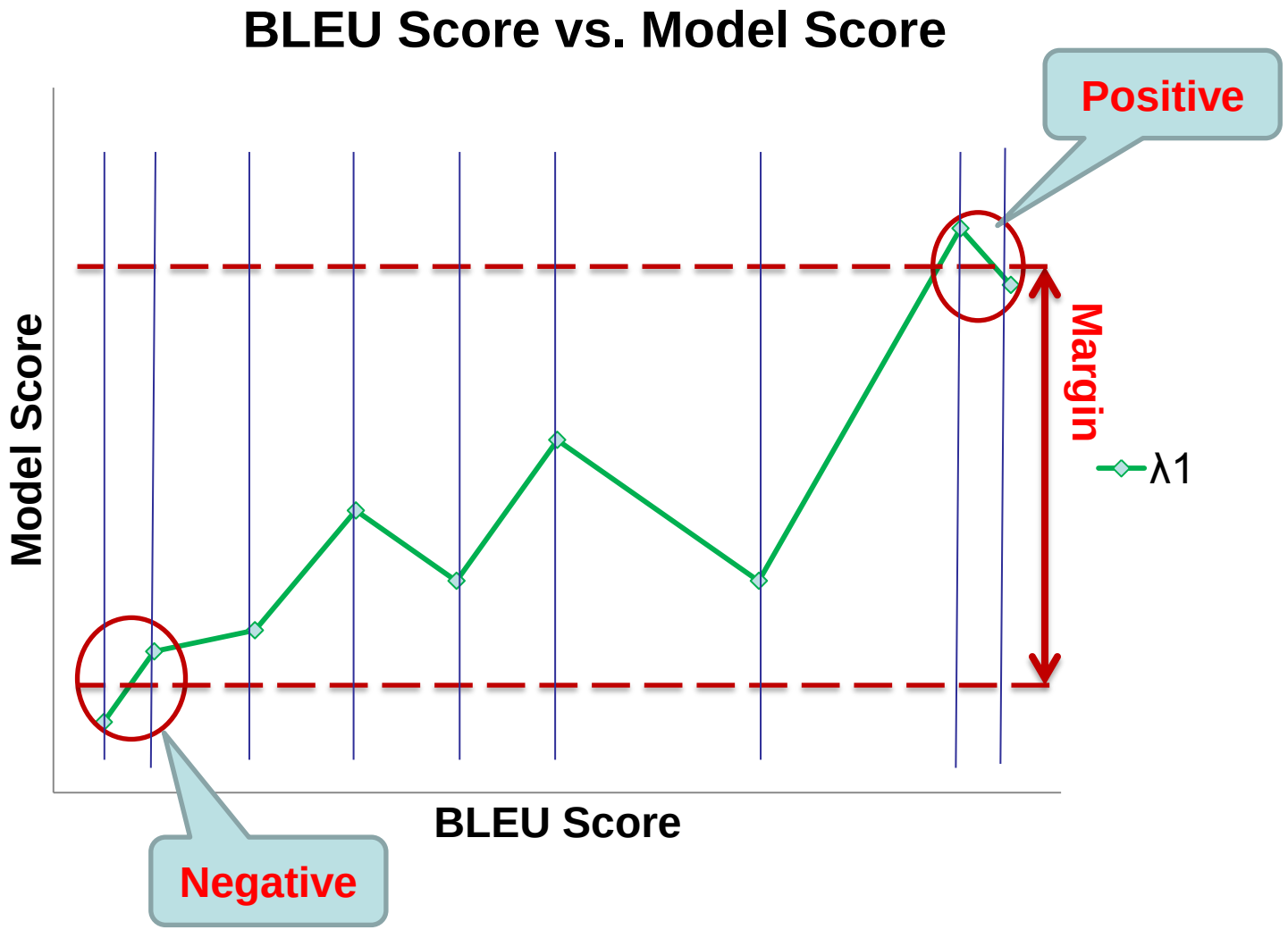
MER Training



Max-Margin Training (MIRA)

- Max-Margin Training focuses the positive candidates and the negative candidates
- Max-Margin Training tries the maximize the margin of the model scores between positive candidates and the negative candidates
- Max-Margin Training does not care about the model scores of the medial candidates

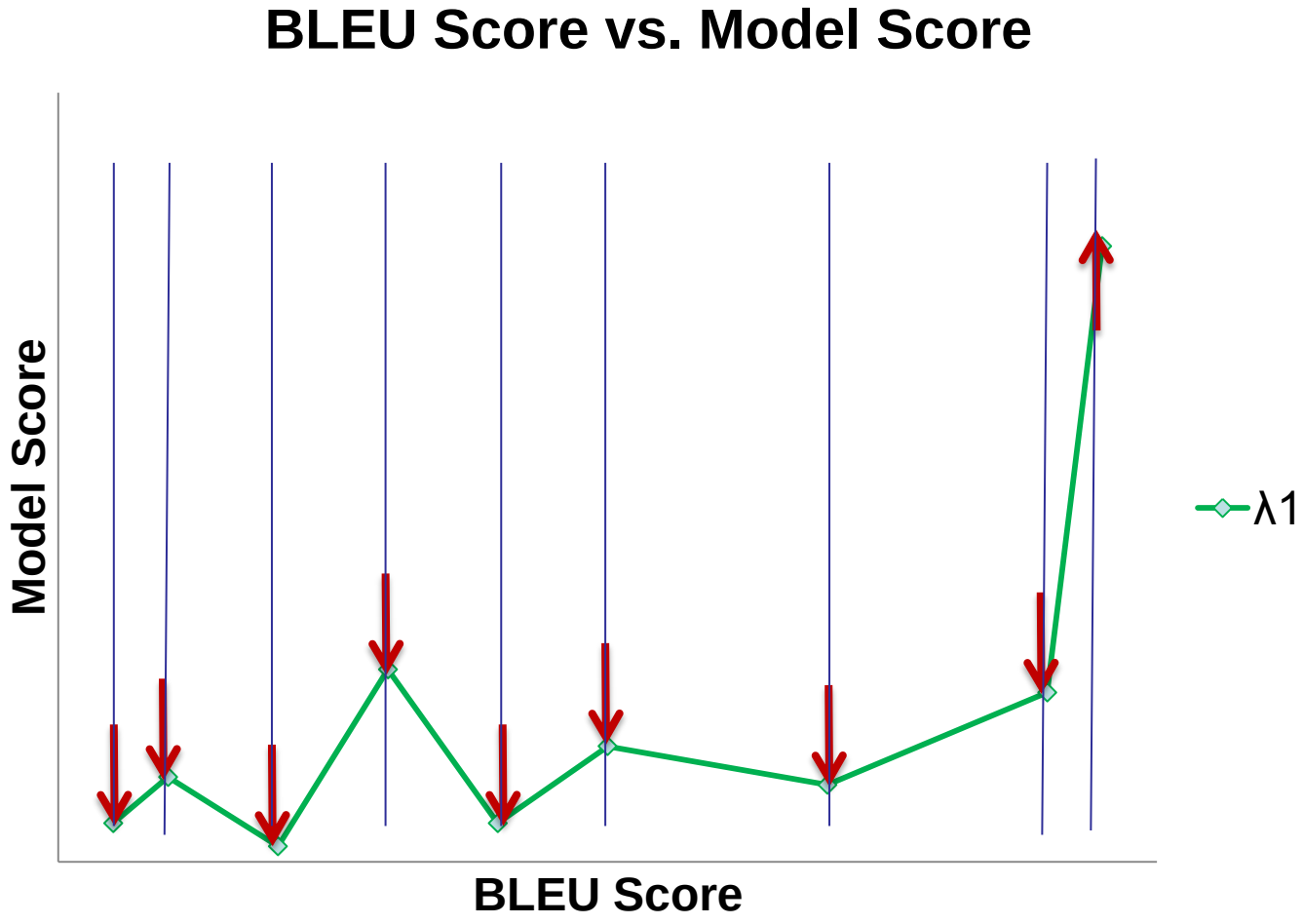
Max-Margin Training (MIRA)



Min-Risk Training

- Min-Risk training tends to maximize the model score of the candidate with the highest BLEU score, while minimize the model scores of all other candidates
- Min-Risk does not care if the line is monotone or not

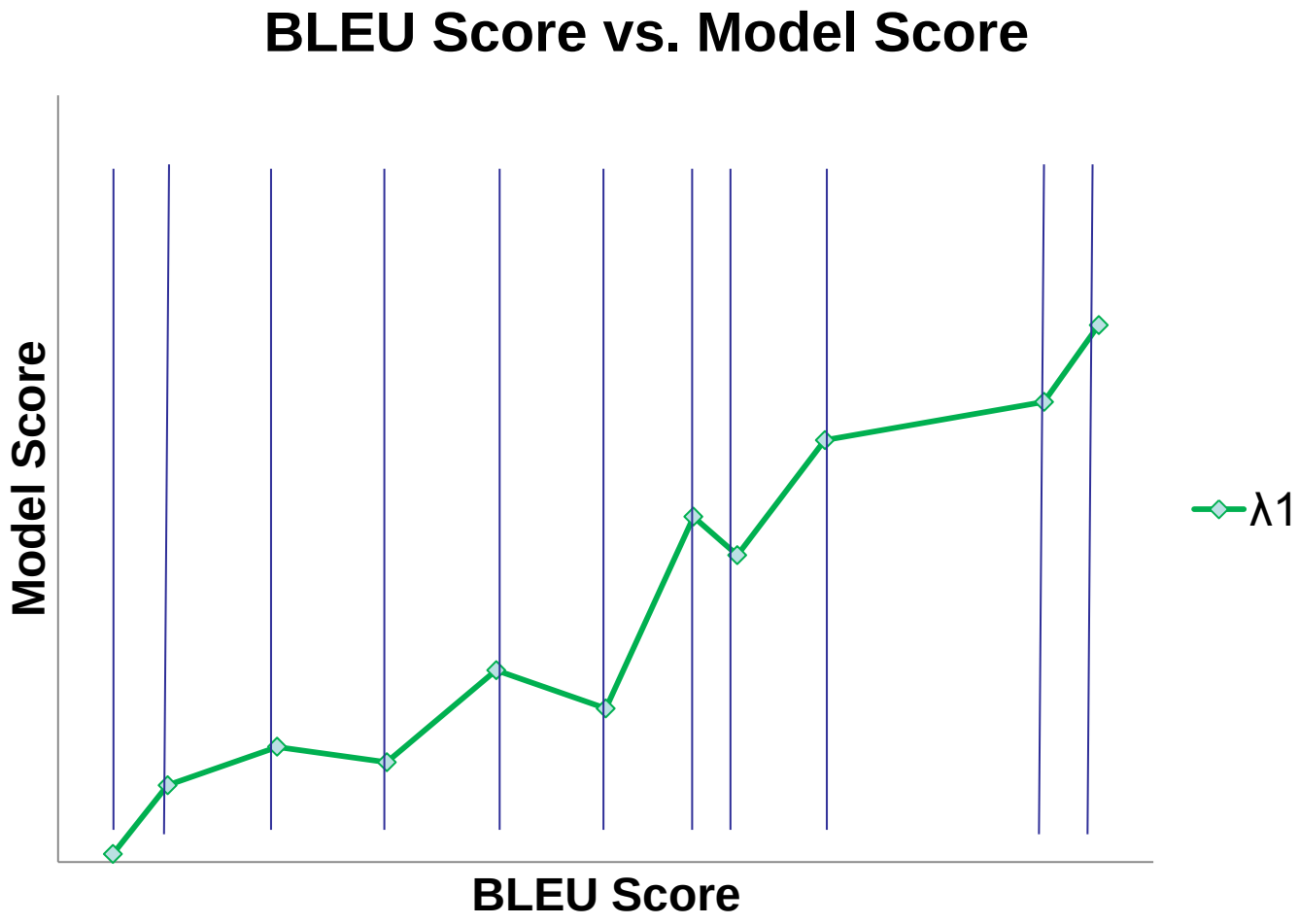
Min-Risk Training



MRC Training

- MRC Training tries to make the whole line most looks monotone
- MRC Training does not ensure the right most dot be the highest one

MRC Training



MERT + MRCT

- MRCT may be regarded as a regularization for MERT
  - There are many possible choices which satisfy the MER criteria, while some of these choices are severely non-monotone
  - The MRCT helps to choose the parameter which most looks monotone, while satisfy the MER criteria
- That’s the reason why:

MERT+MRCT > MERT

Future Question

- Why the improvements of MERT+MRCT on in-domain test data is much larger than that on out-of-domain test data?

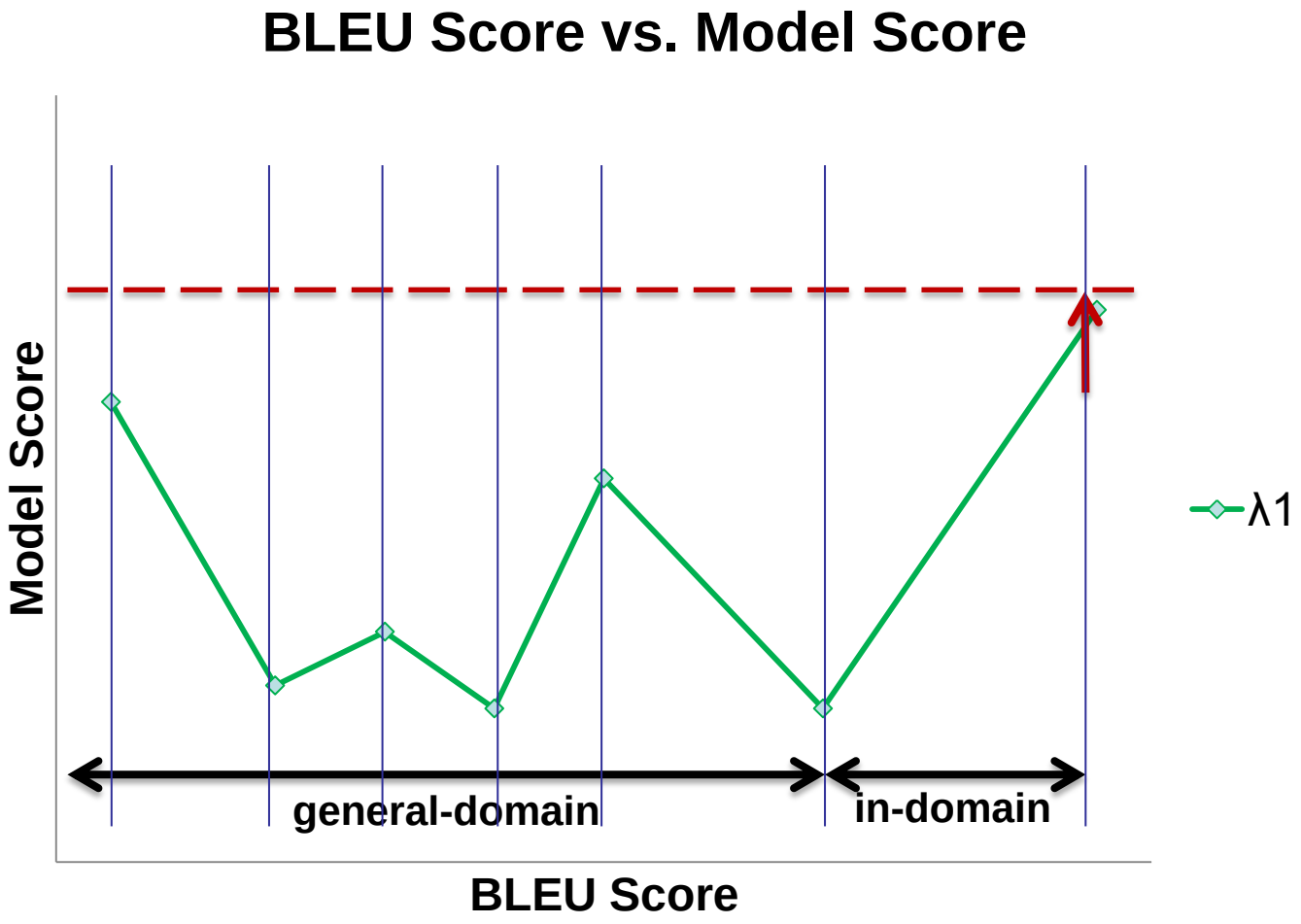
Answer (1/4)

- From the in-domain training data, we obtain both in-domain knowledge and general-domain knowledge.
- In the decoding process, in-domain knowledge and general-domain knowledge are in competition.

Answer (2/4)

- In the n-best list, some candidates are translated using more in-domain knowledge, while some are using more general-domain knowledge.
- The candidates translated using more in-domain knowledge usually get higher BLEU score because the references is given by in-domain development set.

MER Training



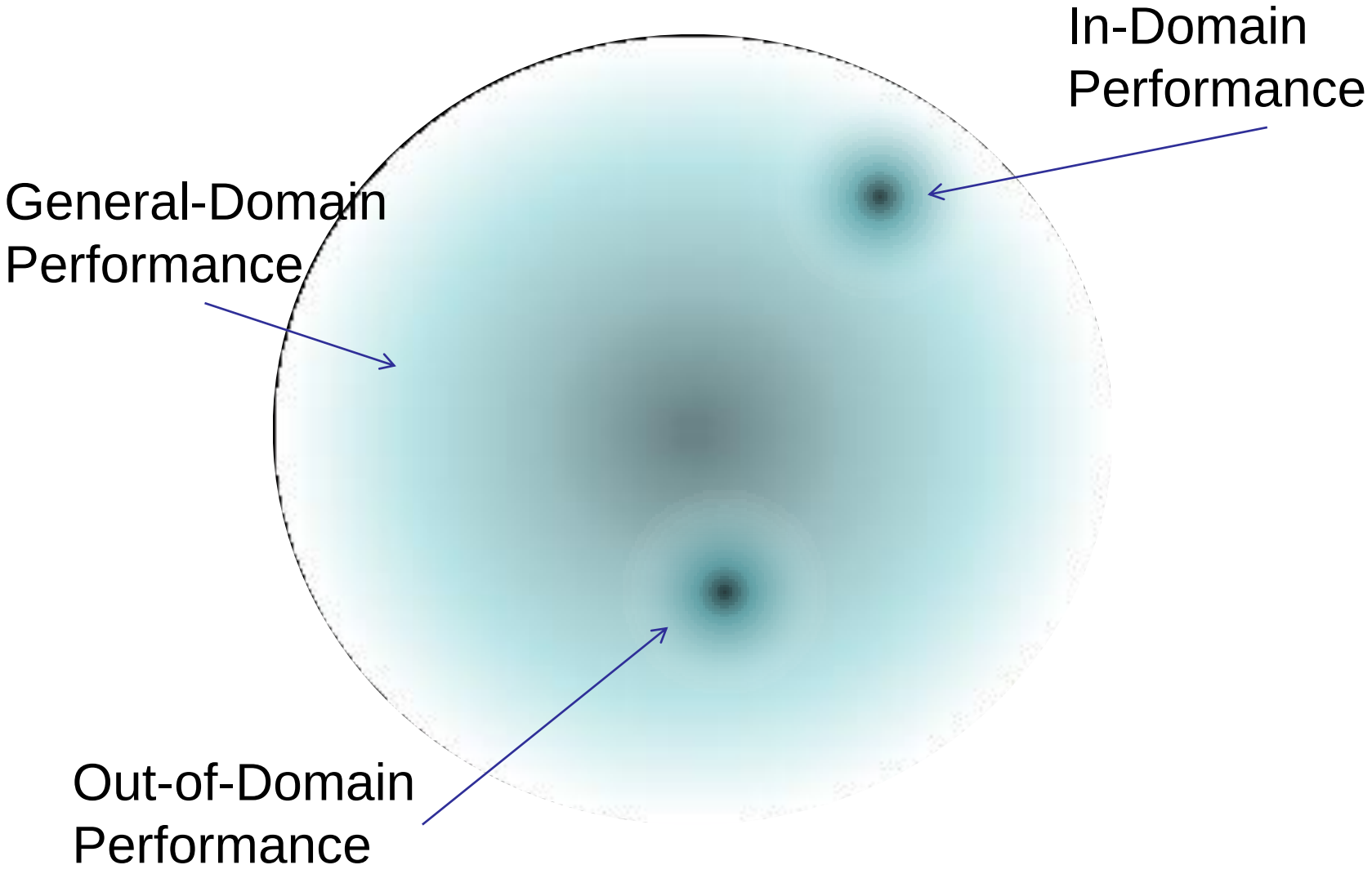
Answer (3/4)

- We may find that the in-domain part of the MERT line is basically monotone, while the general-domain part is not.
- But the MRCT line is almost monotone for all parts.

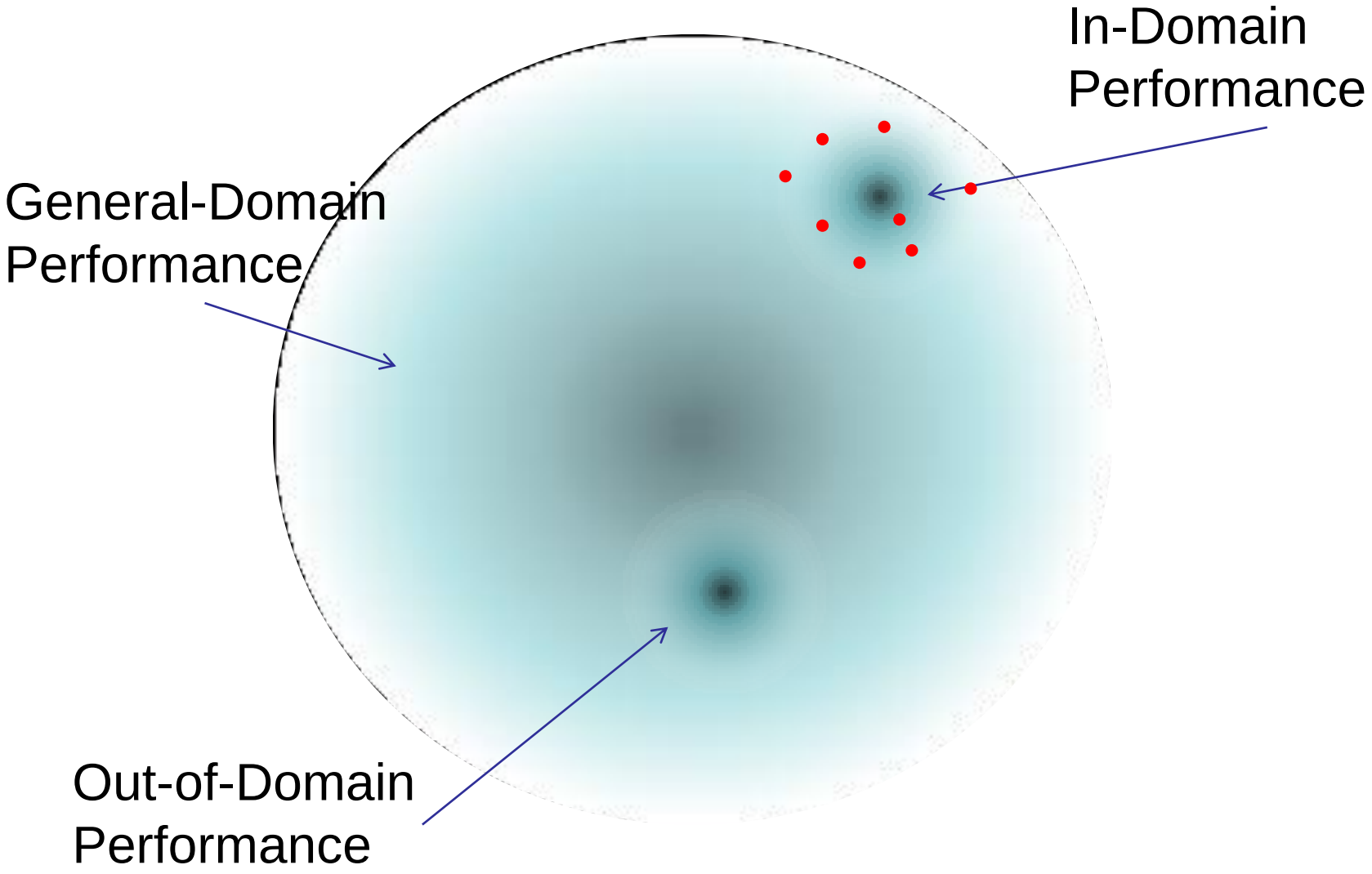
Model Space

- Consider a space consist of all models, where each model is a dot in the space.
- The models perform well in general domain are distributed different with those perform well in specific domains.

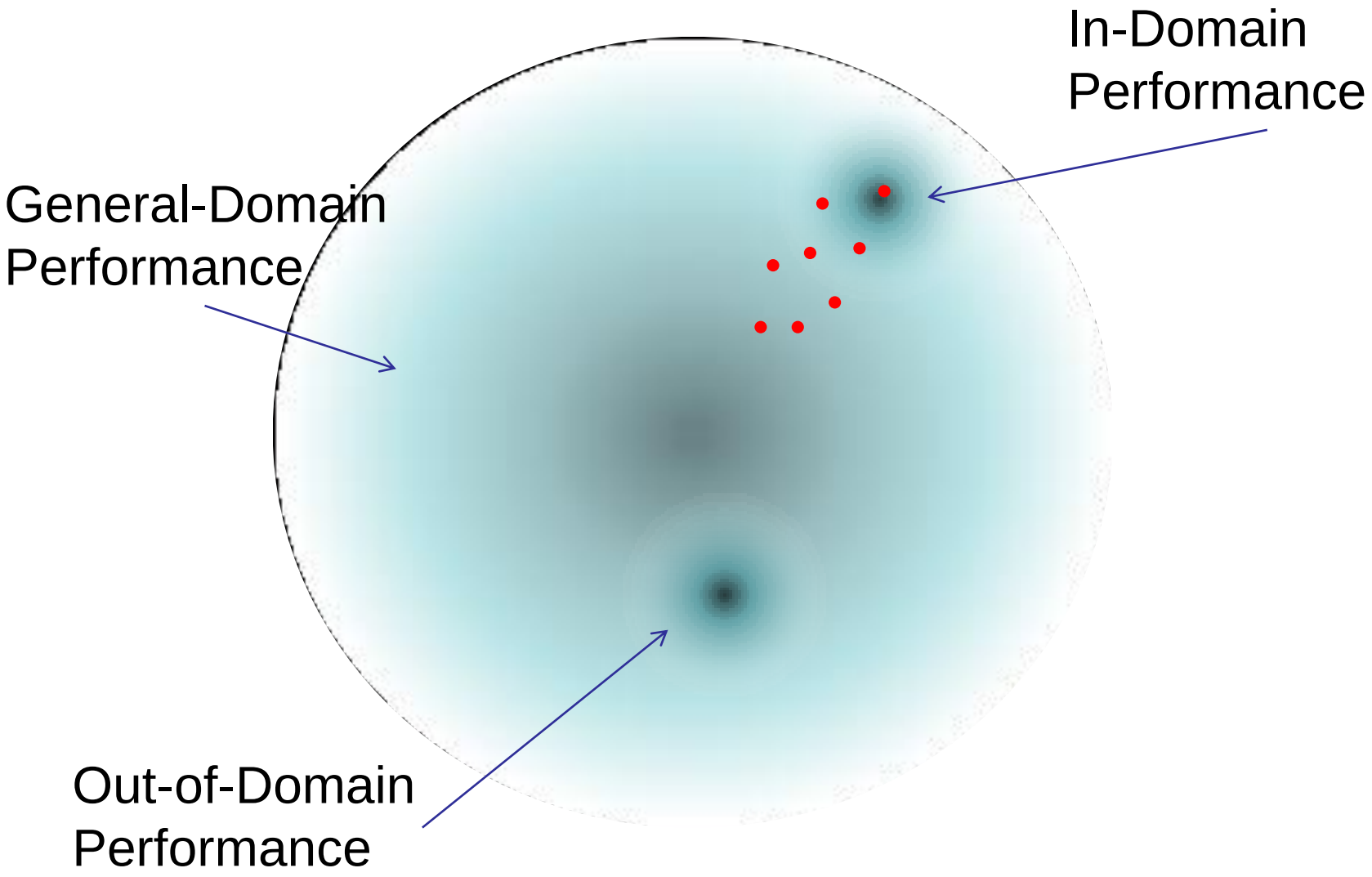
Model Space


In-Domain
Performance
General-Domain
Performance
Out-of-Domain
Performance

MERT


In-Domain
Performance
General-Domain
Performance
Out-of-Domain
Performance

MERT+MRCT


In-Domain
Performance
General-Domain
Performance
Out-of-Domain
Performance

Answer (4/4)

- We can see that the model trained using MERT+MRCT will gain better performance on general-domain test data, as well as on out-of-domain test data, even if we do not the out-of-domain data for training

Outline

- Introduction
- Motivation
- Maximum Rank Correlation Training
- Experiments and Results
- Result Analysis
- Conclusion

Conclusion

- We propose a Maximum Rank Correlation Training approach for parameter tuning for SMT
- We using a multi-objection generative algorithm for parameter tuning
- MRCT + MERT performs a little bit better than MERT for in-domain test data, but much more better for out-of-domain test data
- The time cost of MRCT training is acceptable
- We give an reasonable explanation to the results

THANKS!

Q & A