Large-scale Discriminative n-gram Language Models for Statistical Machine Translation

Zhifei Li and Sanjeev Khudanpur
Johns Hopkins University

Language Model: Data Mismatch

Language Model: Data Mismatch

- A regular language model is trained on well-formed monolingual corpora (e.g., Gigaword)

Language Model: Data Mismatch

- **A regular language model is trained on well-formed monolingual corpora (e.g., Gigaword)**
  - it does not require bilingual data

Language Model: Data Mismatch

- **A regular language model is trained on well-formed monolingual corpora (e.g., Gigaword)**
  - it does not require bilingual data
- **During training, the language model does not see the MT outputs**

Language Model: Data Mismatch

- **A regular language model is trained on well-formed monolingual corpora (e.g., Gigaword)**
  - it does not require bilingual data
- **During training, the language model does not see the MT outputs**
  - But, the LM will be used to rank MT outputs

Language Model: Data Mismatch

- **A regular language model is trained on well-formed monolingual corpora (e.g., Gigaword)**
  - it does not require bilingual data
- **During training, the language model does not see the MT outputs**
  - But, the LM will be used to rank MT outputs
  - MT outputs differ substantially from Gigaword

Language Model: Data Mismatch

- **A regular language model is trained on well-formed monolingual corpora (e.g., Gigaword)**
  - it does not require bilingual data
- **During training, the language model does not see the MT outputs**
  - But, the LM will be used to rank MT outputs
  - MT outputs differ substantially from Gigaword
- **Can we make the LM task-specific without losing its big advantage in using enormous monolingual data?**

Task: reranking MT outputs

Task: reranking MT outputs

我是最好的翻译。

Task: reranking MT outputs

我是最好的翻译。

Task: reranking MT outputs

我是最好的翻译。 → ***I am the best translation.***

Task: reranking MT outputs

我是最好的翻译。 → ***I am the best translation.***

Hypothesized translation	TM	LM	Score
i am a most best translation .	9.1	10	19.1
i are the best translation .	9.0	10	19.0
i am the best translation .	10	8	18
i are the good translate .	9	8	17

Task: reranking MT outputs

我是最好的翻译。 → ***I am the best translation.***

Hypothesized translation	TM	LM	Score
i am a most best translation .	9.1	10	19.1
i are the best translation .	9.0	10	19.0
i am the best translation .	10	8	18
i are the good translate .	9	8	17

Hypothesized translation	TM	LM	Corrective	Score
i am the best translation .	10	8	1.0	19
i am a most best translation .	9.1	10	-0.5	18.6
i are the best translation .	9.0	10	-0.5	18.5
i are the good translate .	9	8	-1	16

Task: reranking MT outputs

我是最好的翻译。 → ***I am the best translation.***

Hypothesized translation	TM	LM	Score
i am a most best translation .	9.1	10	19.1
i are the best translation .	9.0	10	19.0
i am the best translation .	10	8	18
i are the good translate .	9	8	17

Hypothesized translation	TM	LM	Corrective	Score
i am the best translation .	10	8	1.0	19
i am a most best translation .	9.1	10	-0.5	18.6
i are the best translation .	9.0	10	-0.5	18.5
i are the good translate .	9	8	-1	16

Task: reranking MT outputs

我是最好的翻译。 → ***I am the best translation.***

Hypothesized translation	TM	LM	Score
i am a most best translation .	9.1	10	19.1
i are the best translation .	9.0	10	19.0
i am the best translation .	10	8	18
i are the good translate .	9	8	17

Hypothesized translation	TM	LM	Corrective	Score
i am the best translation .	10	8	1.0	19
i am a most best translation .	9.1	10	-0.5	18.6
i are the best translation .	9.0	10	-0.5	18.5
i are the good translate .	9	8	-1	16

Task: reranking MT outputs

我是最好的翻译。 → ***I am the best translation.***

Hypothesized translation	TM	LM	Score
i am a most best translation .	9.1	10	19.1
i are the best translation .	9.0	10	19.0
i am the best translation .	10	8	18
i are the good translate .	9	8	17

Hypothesized translation	TM	LM	Corrective	Score
i am the best translation .	10	8	1.0	19
i am a most best translation .	9.1	10	-0.5	18.6
i are the best translation .	9.0	10	-0.5	18.5
i are the good translate .	9	8	-1	16

Discriminative LM reranking

Discriminative LM reranking

- **A discriminative language model should**

Discriminative LM reranking

- **A discriminative language model should**
  - discover useful n-gram features

Discriminative LM reranking

- **A discriminative language model should**
  - discover useful n-gram features
  - find optimal weights for these features

Discriminative LM reranking

- **A discriminative language model should**
  - discover useful n-gram features
  - find optimal weights for these features
- **The discriminative LM is trained on**

Discriminative LM reranking

- **A discriminative language model should**
  - discover useful n-gram features
  - find optimal weights for these features
- **The discriminative LM is trained on**
  - hypotheses produced by a baseline system

Discriminative LM reranking

- **A discriminative language model should**
  - discover useful n-gram features
  - find optimal weights for these features
- **The discriminative LM is trained on**
  - hypotheses produced by a baseline system
  - desired translation

Discriminative Modeling

Discriminative Modeling

- **Global linear model**

$$s(f, e) = \Phi(f, e) \cdot \vec{\alpha} = \sum_j \Phi_j(f, e)\alpha_j$$

Discriminative Modeling

- **Global linear model**

$$s(f,e) = \Phi(f,e) \cdot \vec{\alpha} = \sum_j \Phi_j(f,e)\alpha_j$$

- **Training**

$$\vec{\alpha}^* = \arg\max_{\vec{\alpha}} F(Data, \vec{\alpha})$$

Discriminative Modeling

- **Global linear model**

$$s(f,e) = \Phi(f,e) \cdot \vec{\alpha} = \sum_j \Phi_j(f,e)\alpha_j$$

- **Training**

Perceptron

$$\vec{\alpha}^* = \arg\max_{\vec{\alpha}} F(Data, \vec{\alpha})$$

Discriminative Modeling

- **Global linear model**

$$s(f,e) = \Phi(f,e) \cdot \vec{\alpha} = \sum_j \Phi_j(f,e)\alpha_j$$

- **Training**

$$\vec{\alpha}^* = \arg\max_{\vec{\alpha}} F(Data, \vec{\alpha})$$

Perceptron

CRF

Discriminative Modeling

- **Global linear model**

$$s(f,e) = \Phi(f,e) \cdot \vec{\alpha} = \sum_j \Phi_j(f,e)\alpha_j$$

- **Training**

$$\vec{\alpha}^* = \arg\max_{\vec{\alpha}} F(Data, \vec{\alpha})$$

Perceptron

CRF

Min Risk

Discriminative Modeling

- **Global linear model**

$$s(f,e) = \Phi(f,e) \cdot \vec{\alpha} = \sum_j \Phi_j(f,e)\alpha_j$$

- **Training**

$$\vec{\alpha}^* = \arg\max_{\vec{\alpha}} F(Data, \vec{\alpha})$$

Perceptron

CRF

Min Risk

Max Margin

Discriminative Modeling

- **Global linear model**

$$s(f,e) = \Phi(f,e) \cdot \vec{\alpha} = \sum_j \Phi_j(f,e)\alpha_j$$

- **Training**

$$\vec{\alpha}^* = \arg\max_{\vec{\alpha}} F(Data, \vec{\alpha})$$

Perceptron

CRF

Min Risk

Max Margin

- **Decision rule**

$$e^* = \arg\max_{e \in \mathrm{TRANS}(f)} s(f,e)$$

Discriminative Reranking

- Score after reranking
- Features

Discriminative Reranking

- **Score after reranking**

$$s(f, e) \quad = \quad \Phi(f, e) \cdot \vec{\alpha}$$

- **Features**

Discriminative Reranking

- **Score after reranking**

$$\begin{aligned} s(f,e) &= \Phi(f,e) \cdot \vec{\alpha} \\ &= \alpha_0 \Phi_0(f,e) + \end{aligned}$$

- **Features**
  - *baseline feature*

$$\Phi_0(f,e) = \text{Baseline score for translation } e$$

Discriminative Reranking

- **Score after reranking**

$$
\begin{aligned}
s(f,e) &= \Phi(f,e)\cdot\vec{\alpha} \\
&= \alpha_0\Phi_0(f,e) + \sum_{j\in[1,J]} \alpha_j\Phi_j(f,e)
\end{aligned}
$$

- **Features**
  - *baseline feature*

$$\Phi_0(f,e) = \text{Baseline score for translation } e$$

  - *reranking n-gram features, e.g.,*

Discriminative Reranking

- **Score after reranking**

$$
\begin{aligned}
s(f,e) &= \Phi(f,e)\cdot\vec{\alpha} \\
&= \alpha_0\Phi_0(f,e) + \sum_{j\in[1,J]} \alpha_j\Phi_j(f,e)
\end{aligned}
$$

- **Features**
  - *baseline feature*

$$\Phi_0(f,e) = \text{Baseline score for translation } e$$

  - *reranking n-gram features, e.g.,*

$$\Phi_1(f,e) = \text{Count of the bigram “}\texttt{the of}\text{” in } e$$

Discriminative Reranking

- **Score after reranking**

$$
\begin{aligned}
s(f,e) &= \Phi(f,e)\cdot\vec{\alpha} \\
&= \alpha_0\Phi_0(f,e) + \sum_{j\in[1,J]} \alpha_j\Phi_j(f,e)
\end{aligned}
$$

- **Features**
  - *baseline feature*

contains a LM score

$$\Phi_0(f,e) = \text{Baseline score for translation } e$$

  - *reranking n-gram features, e.g.,*

$$\Phi_1(f,e) = \text{Count of the bigram “}\texttt{the of}\text{” in } e$$

Discriminative Reranking

- **Score after reranking**

$$
\begin{aligned}
s(f,e) &= \Phi(f,e)\cdot\vec{\alpha} \\
&= \alpha_0\Phi_0(f,e) + \sum_{j\in[1,J]} \alpha_j\Phi_j(f,e)
\end{aligned}
$$

corrective score

- **Features**
  - *baseline feature*

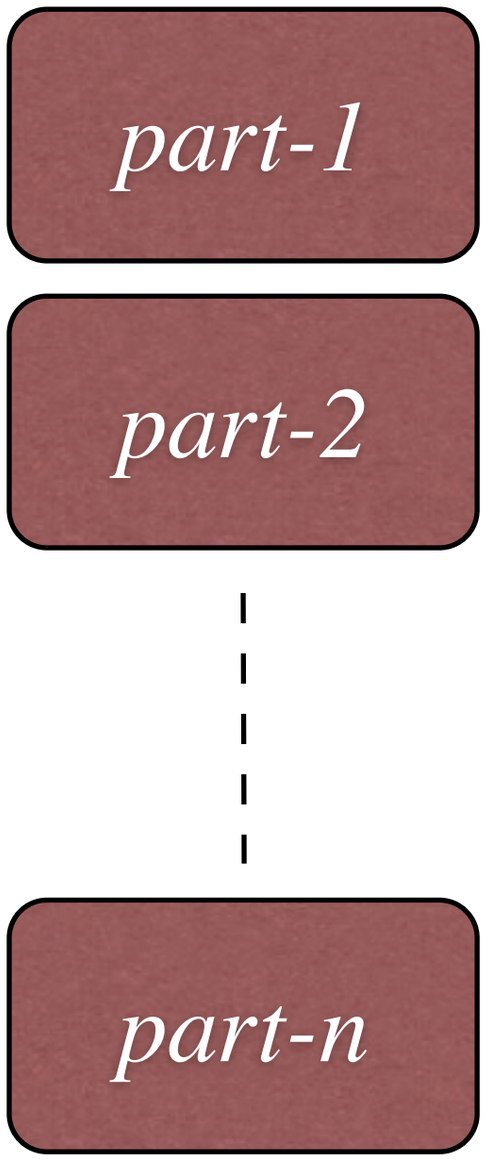
$$\Phi_0(f,e) = \text{Baseline score for translation } e$$

  - *reranking n-gram features, e.g.,*

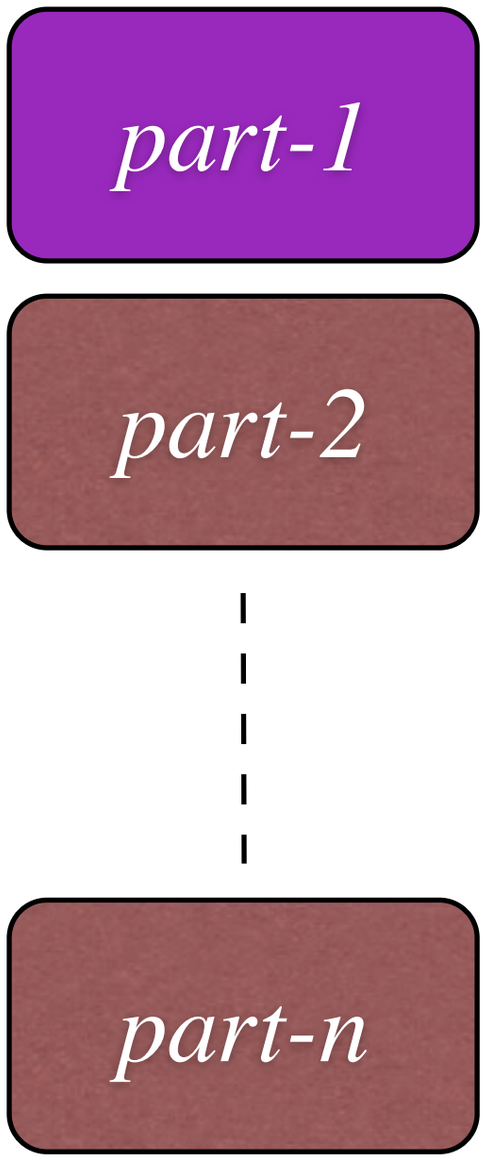
$$\Phi_1(f,e) = \text{Count of the bigram “}\texttt{the of}\text{” in } e$$

Leave-one-out Baseline Training

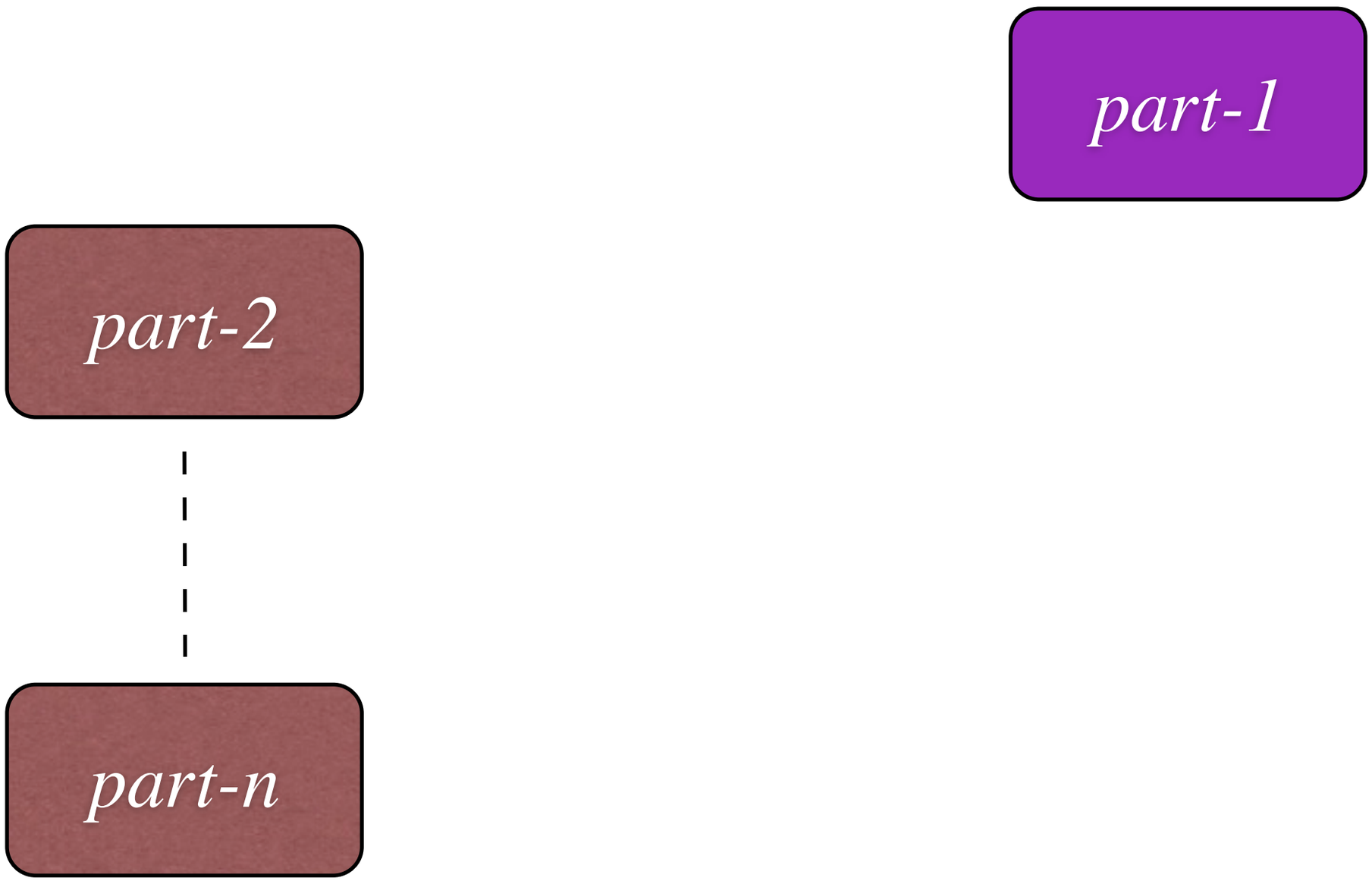
Leave-one-out Baseline Training



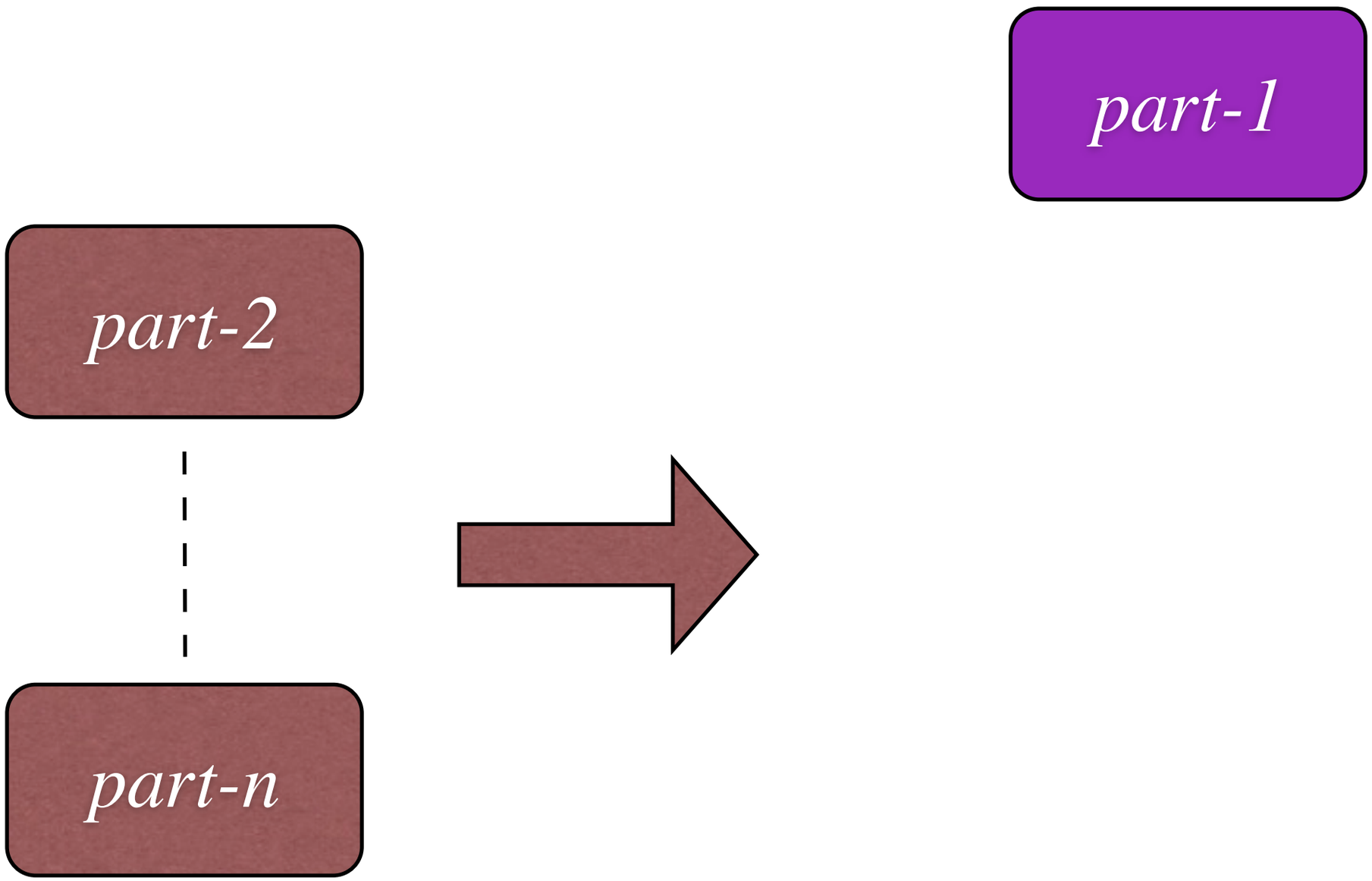
Leave-one-out Baseline Training



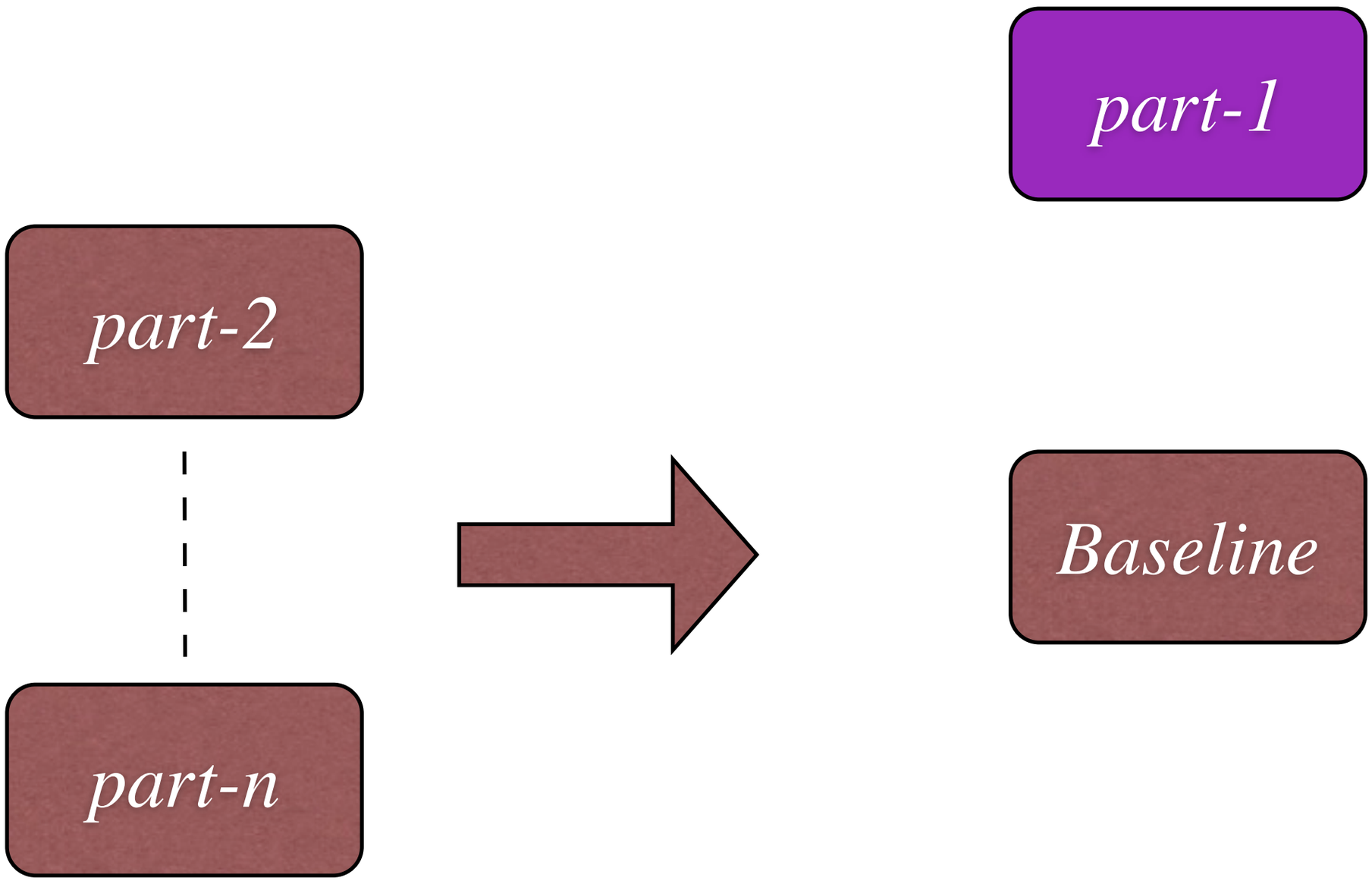
Leave-one-out Baseline Training



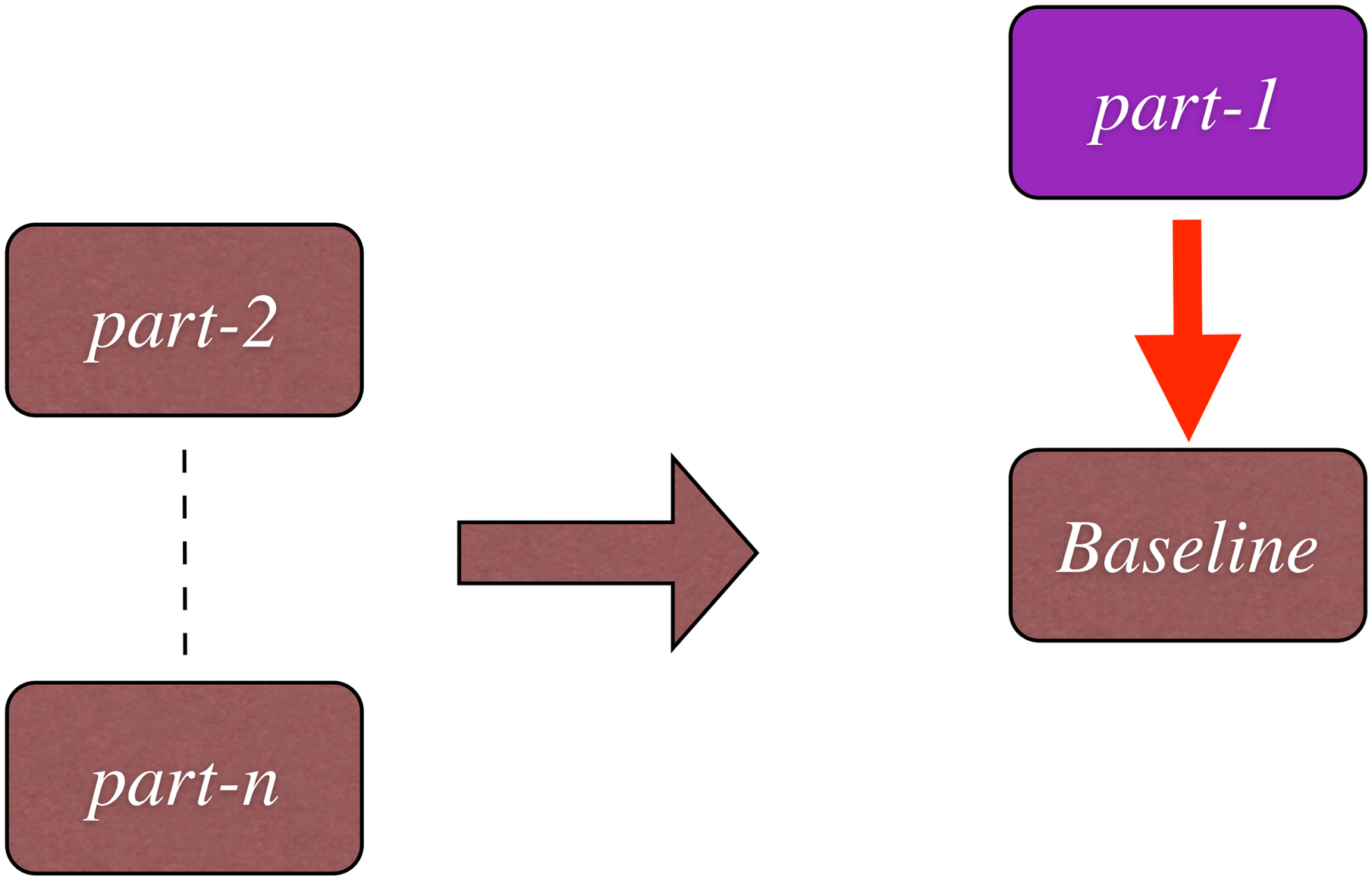
Leave-one-out Baseline Training



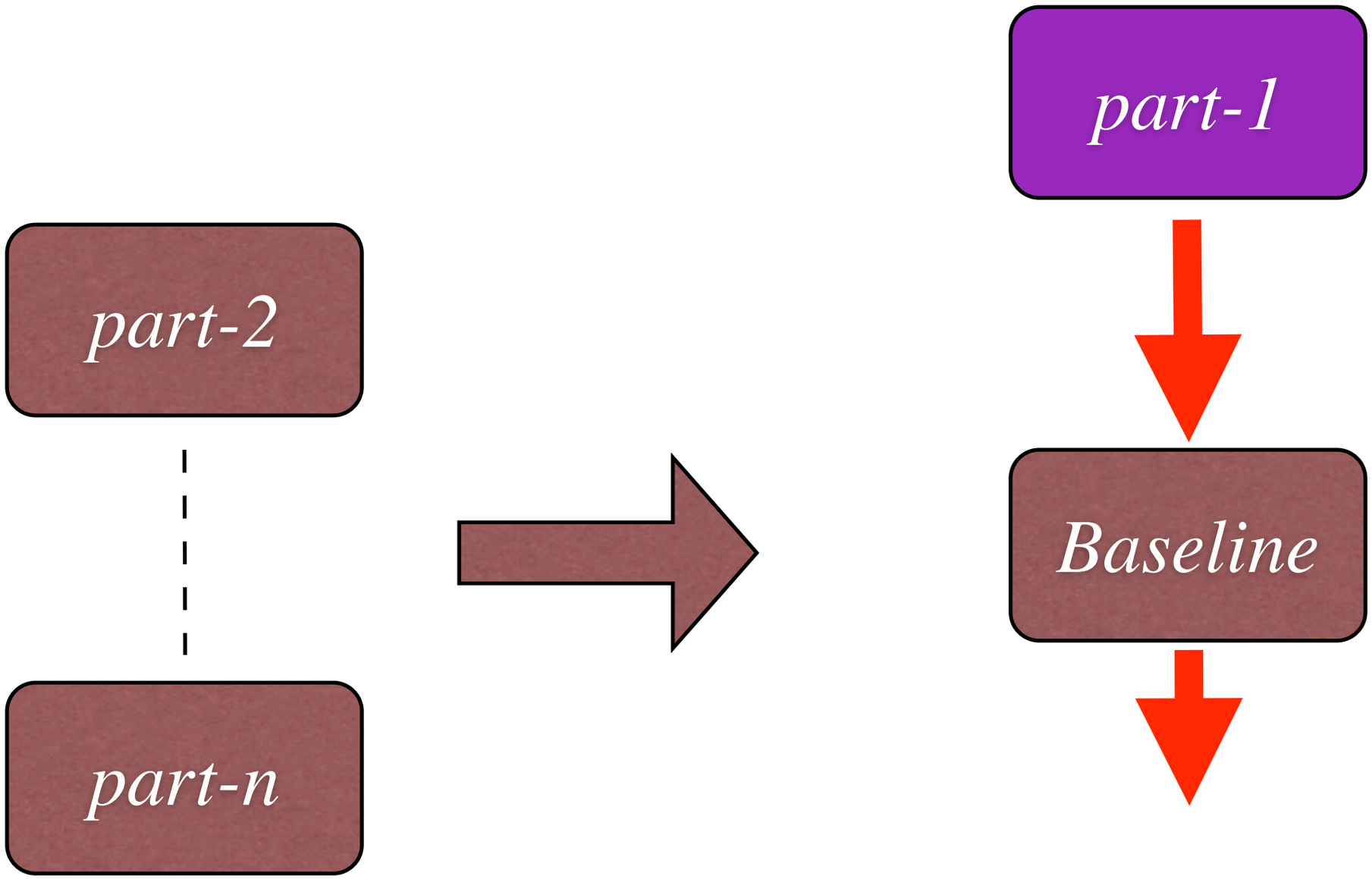
Leave-one-out Baseline Training



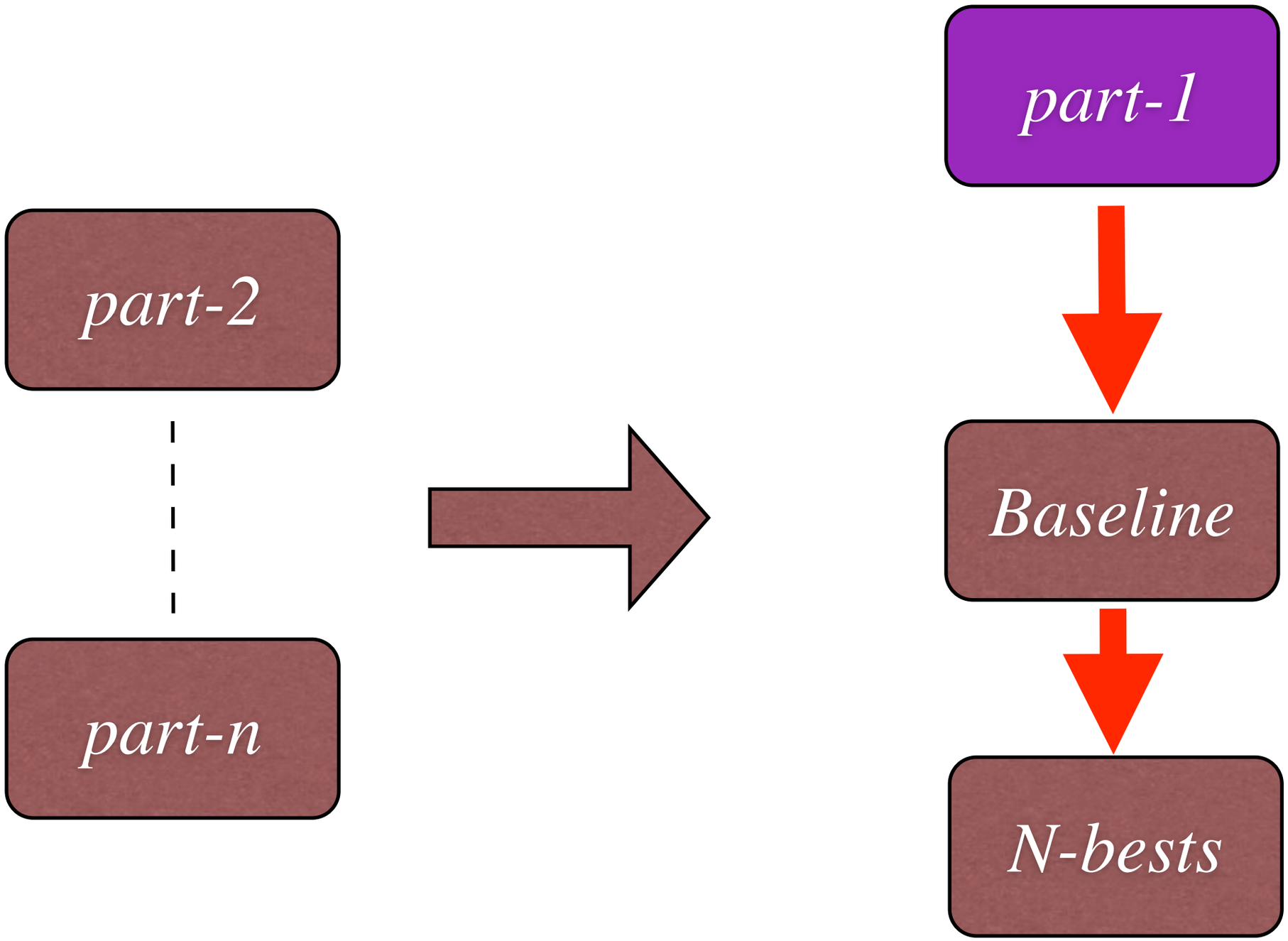
Leave-one-out Baseline Training



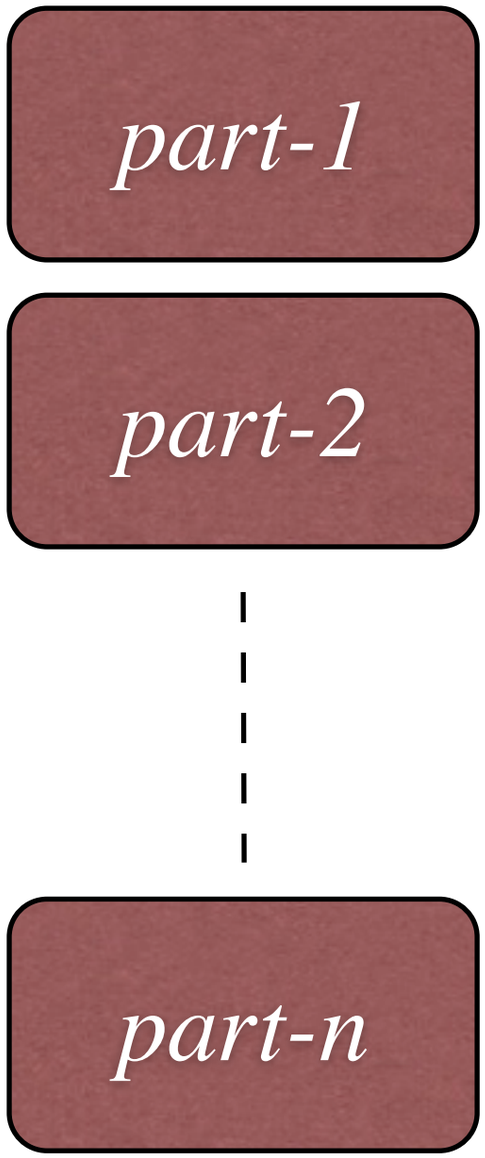
Leave-one-out Baseline Training



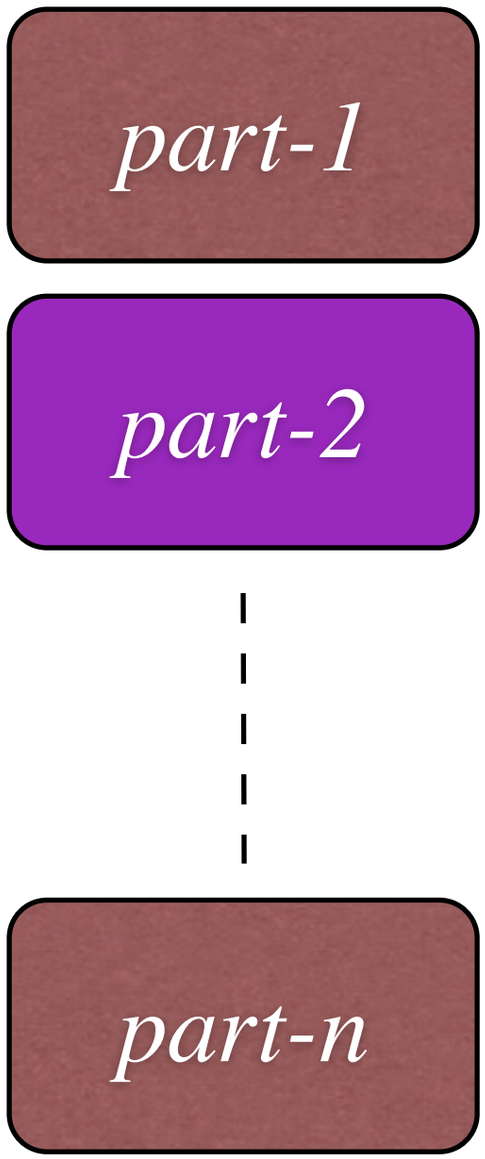
Leave-one-out Baseline Training



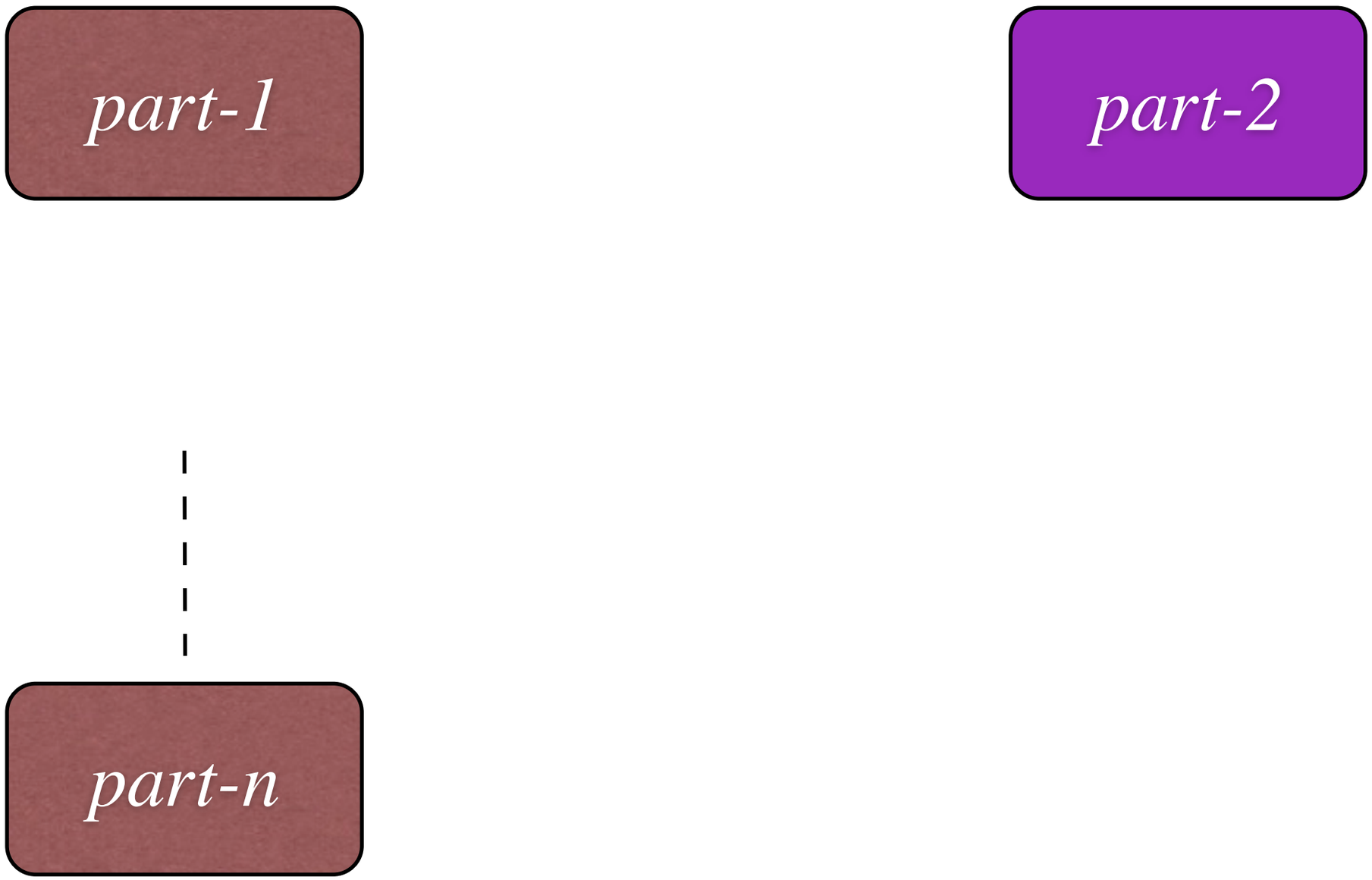
Leave-one-out Baseline Training



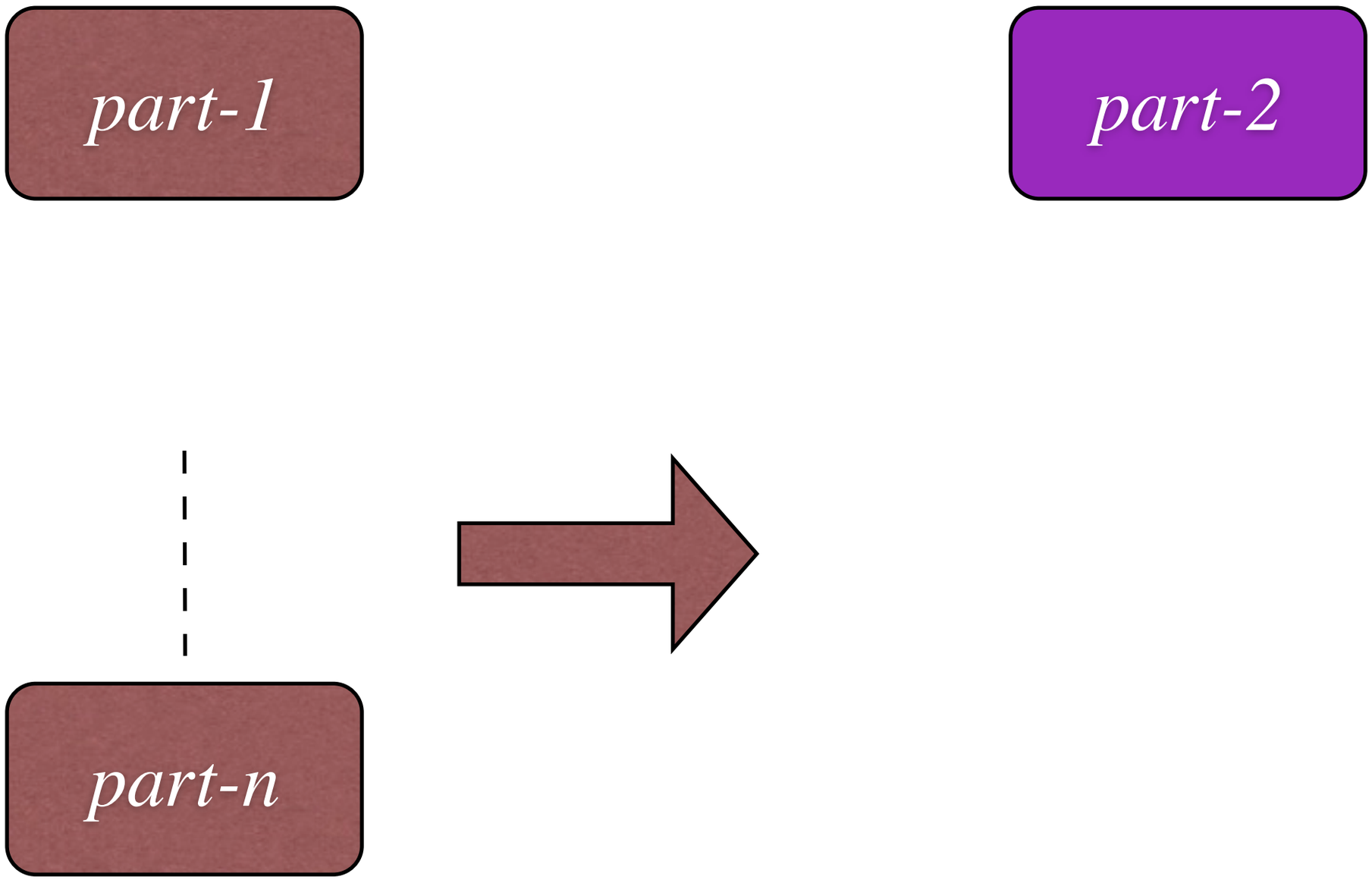
Leave-one-out Baseline Training



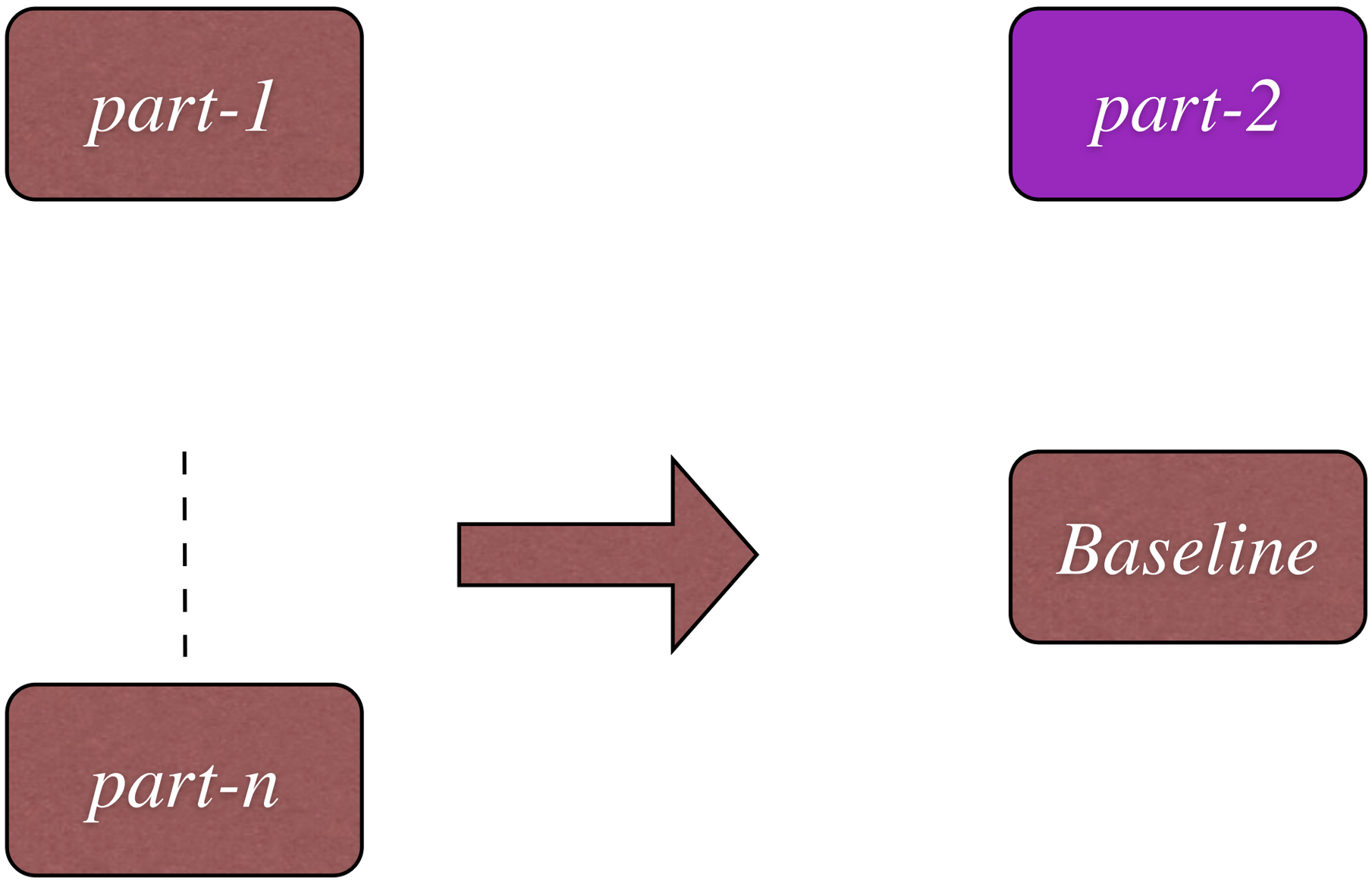
Leave-one-out Baseline Training



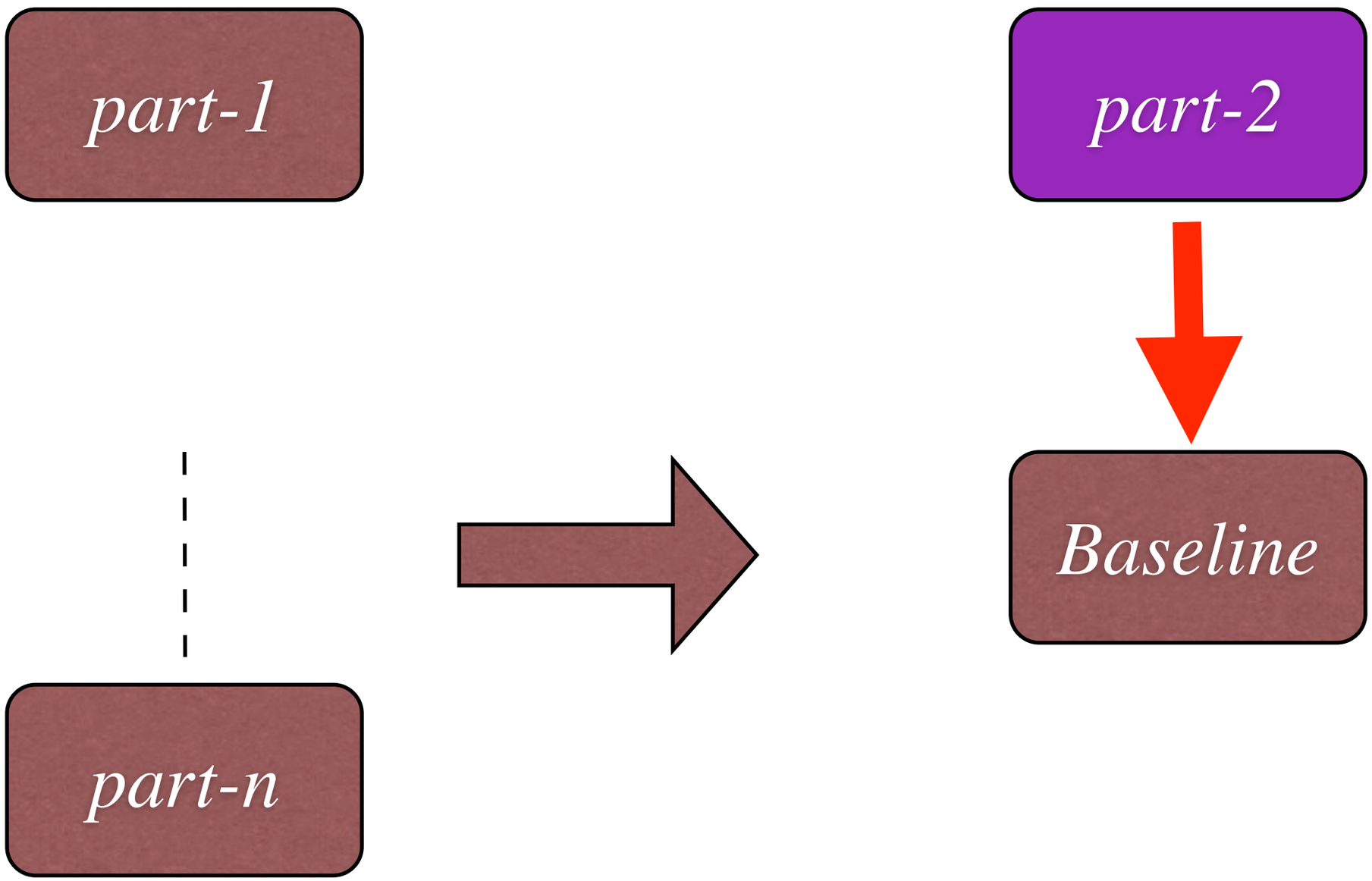
Leave-one-out Baseline Training



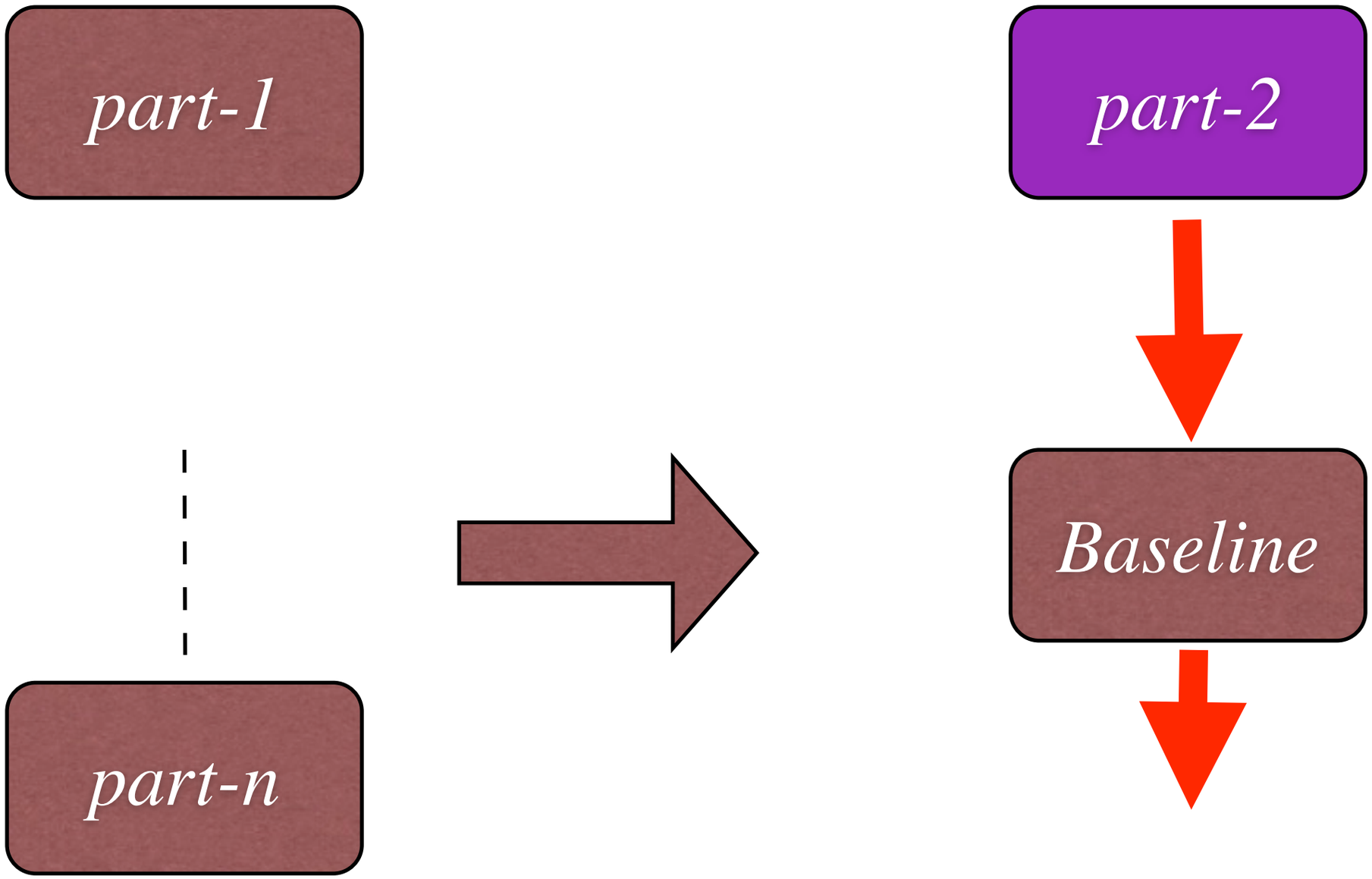
Leave-one-out Baseline Training



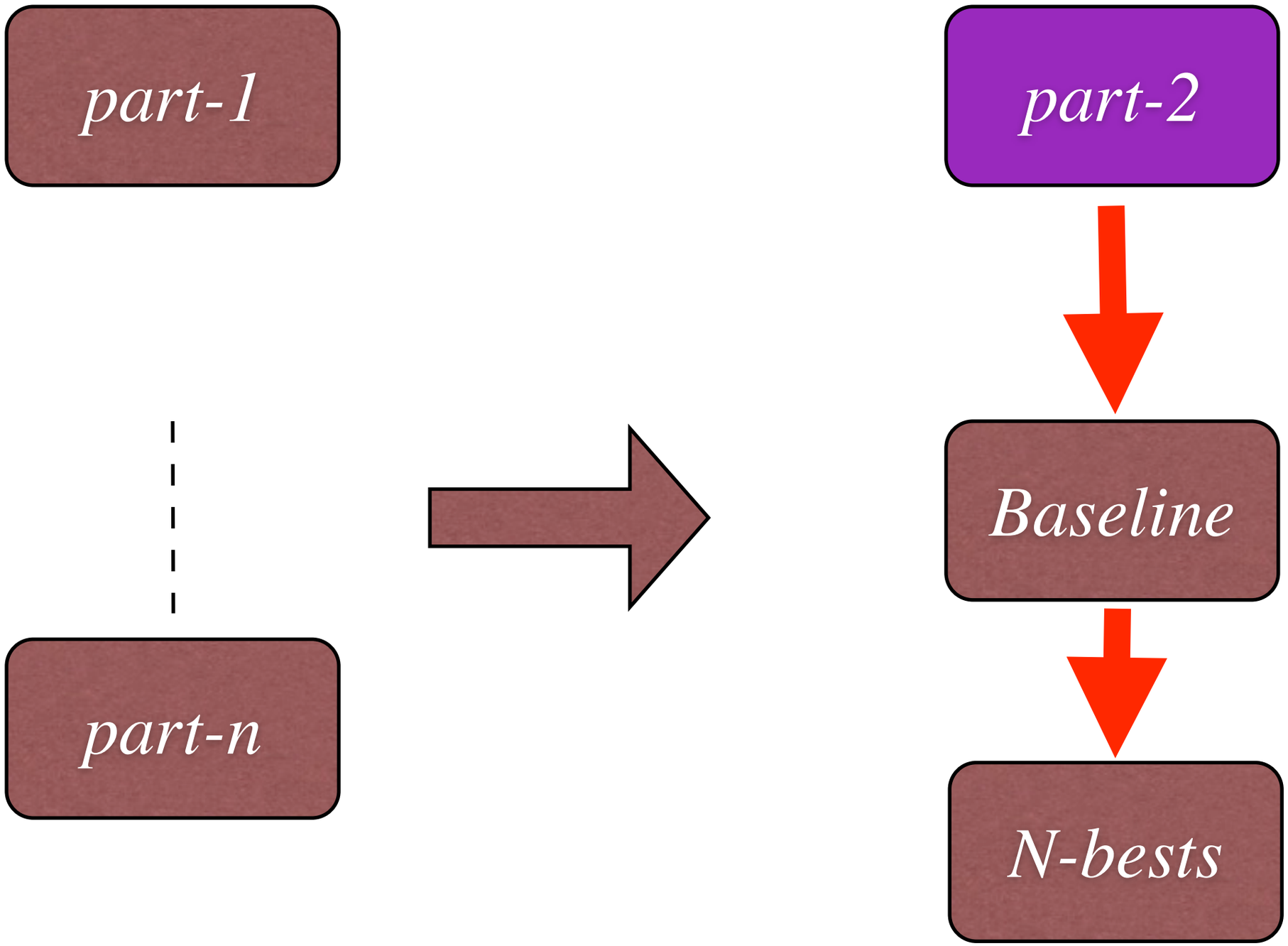
Leave-one-out Baseline Training



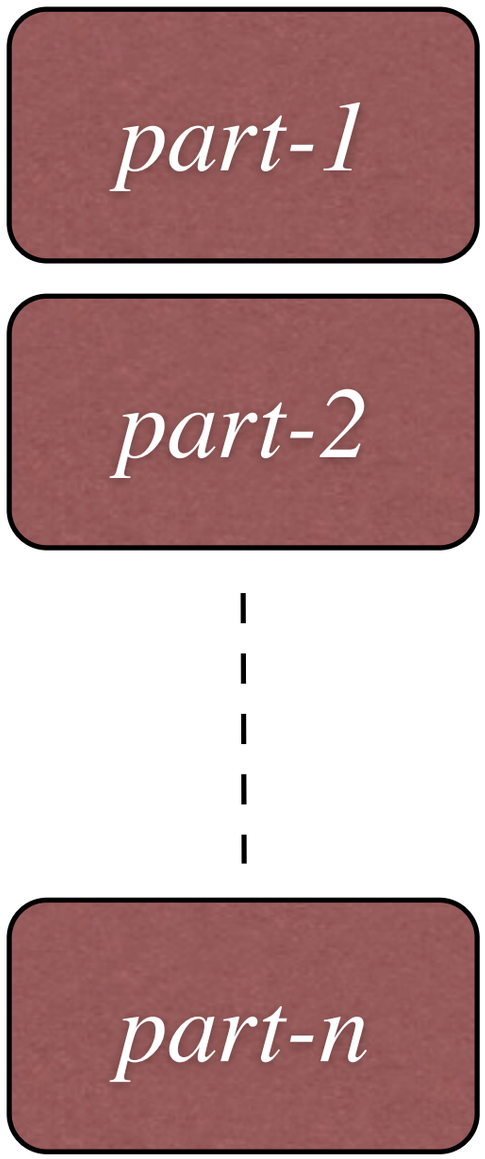
Leave-one-out Baseline Training



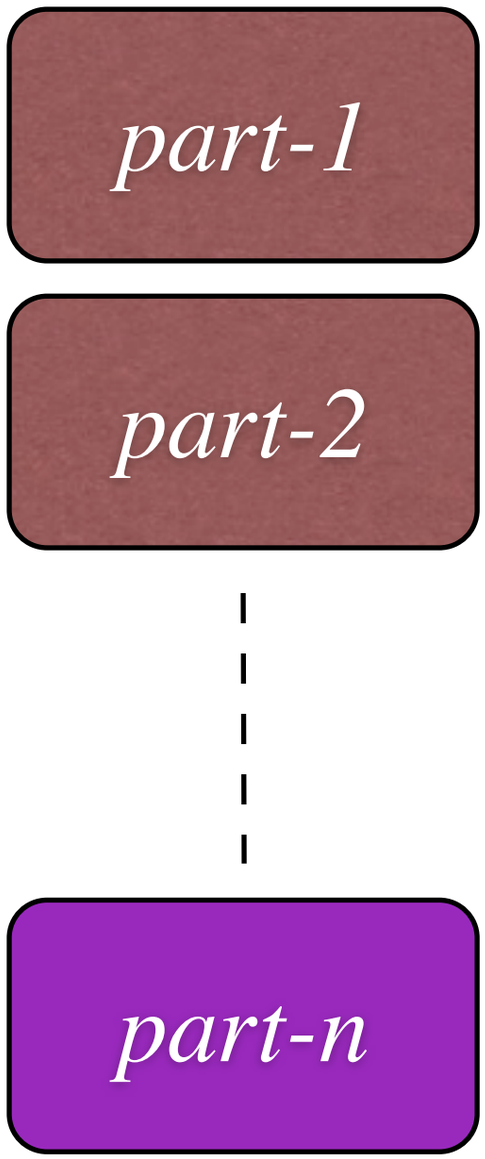
Leave-one-out Baseline Training



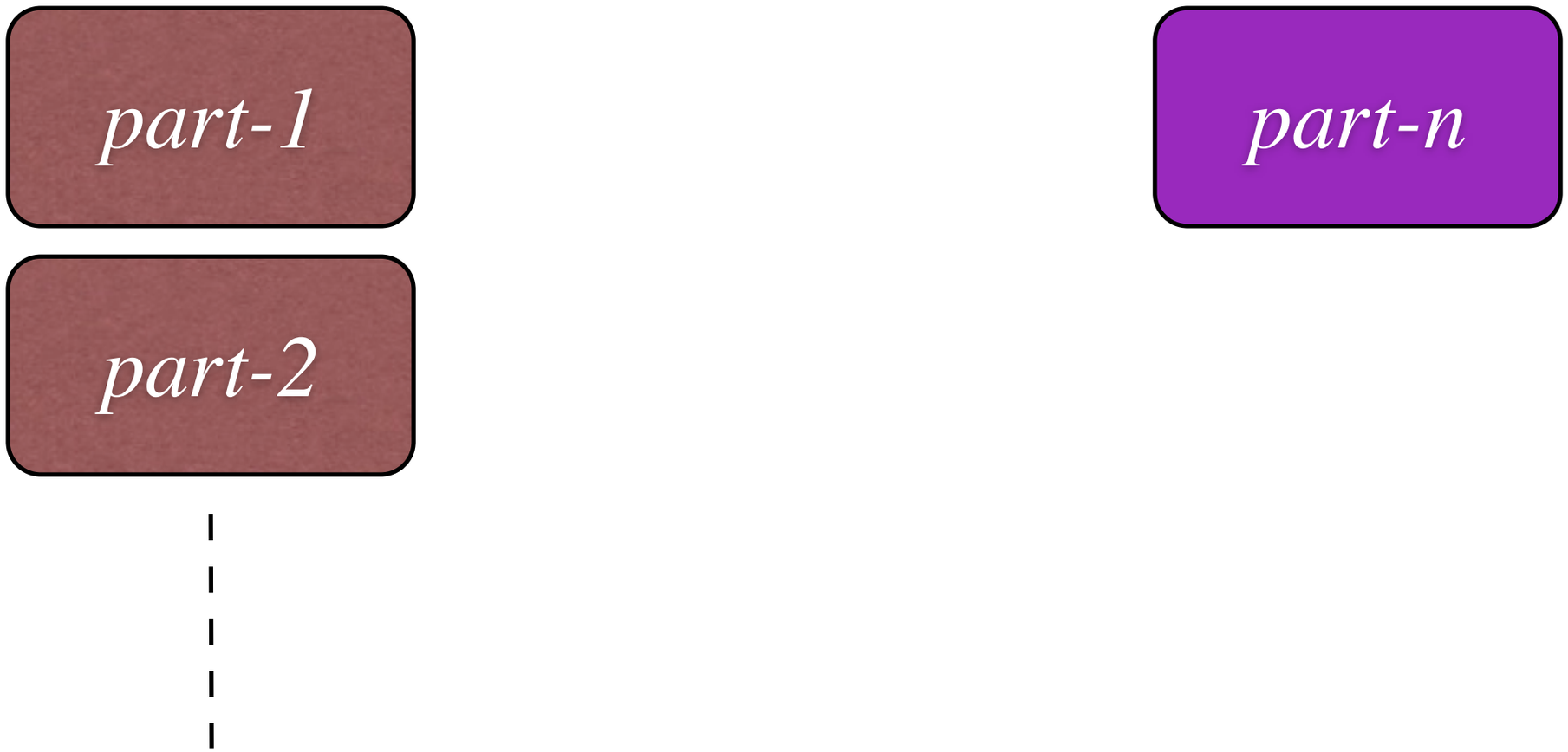
Leave-one-out Baseline Training



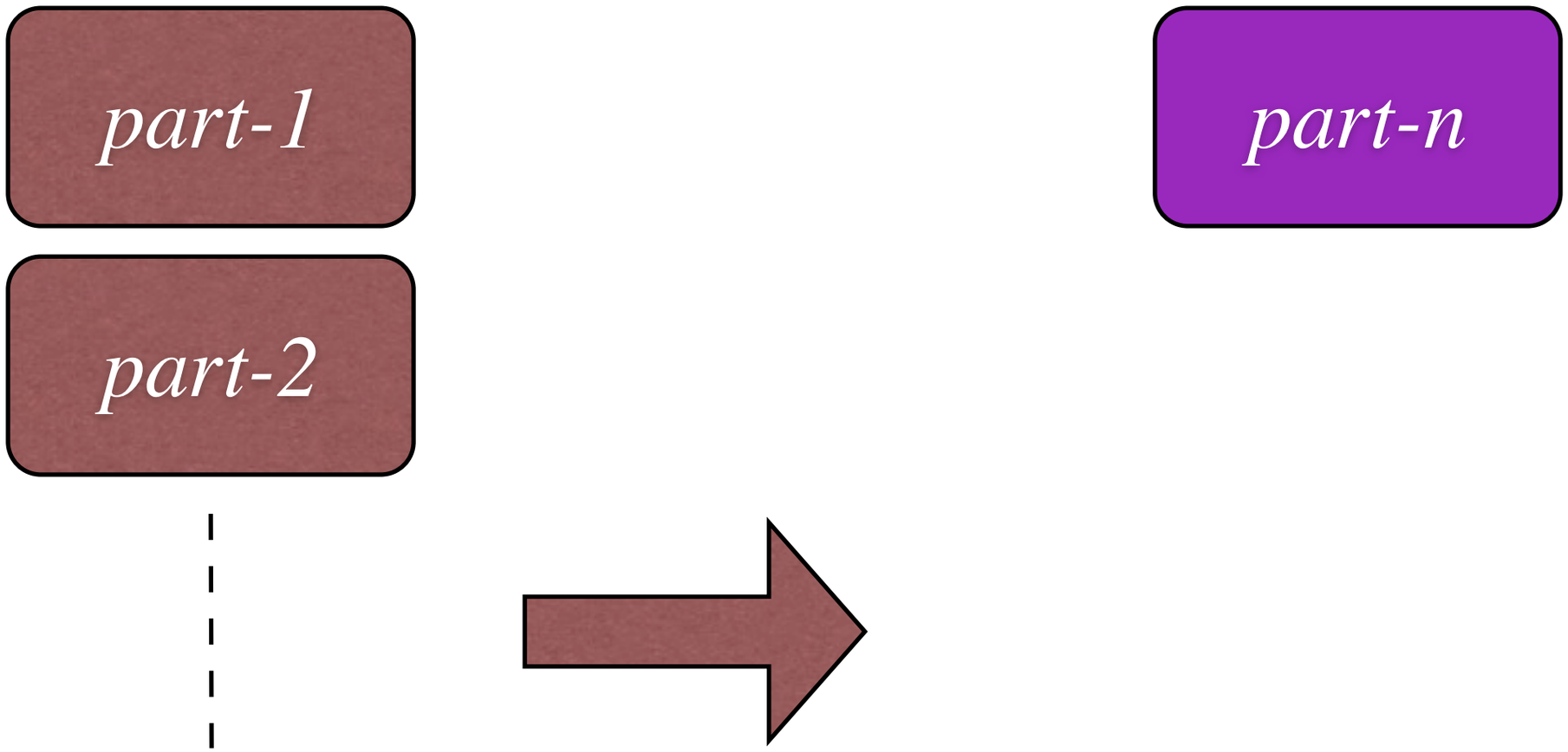
Leave-one-out Baseline Training



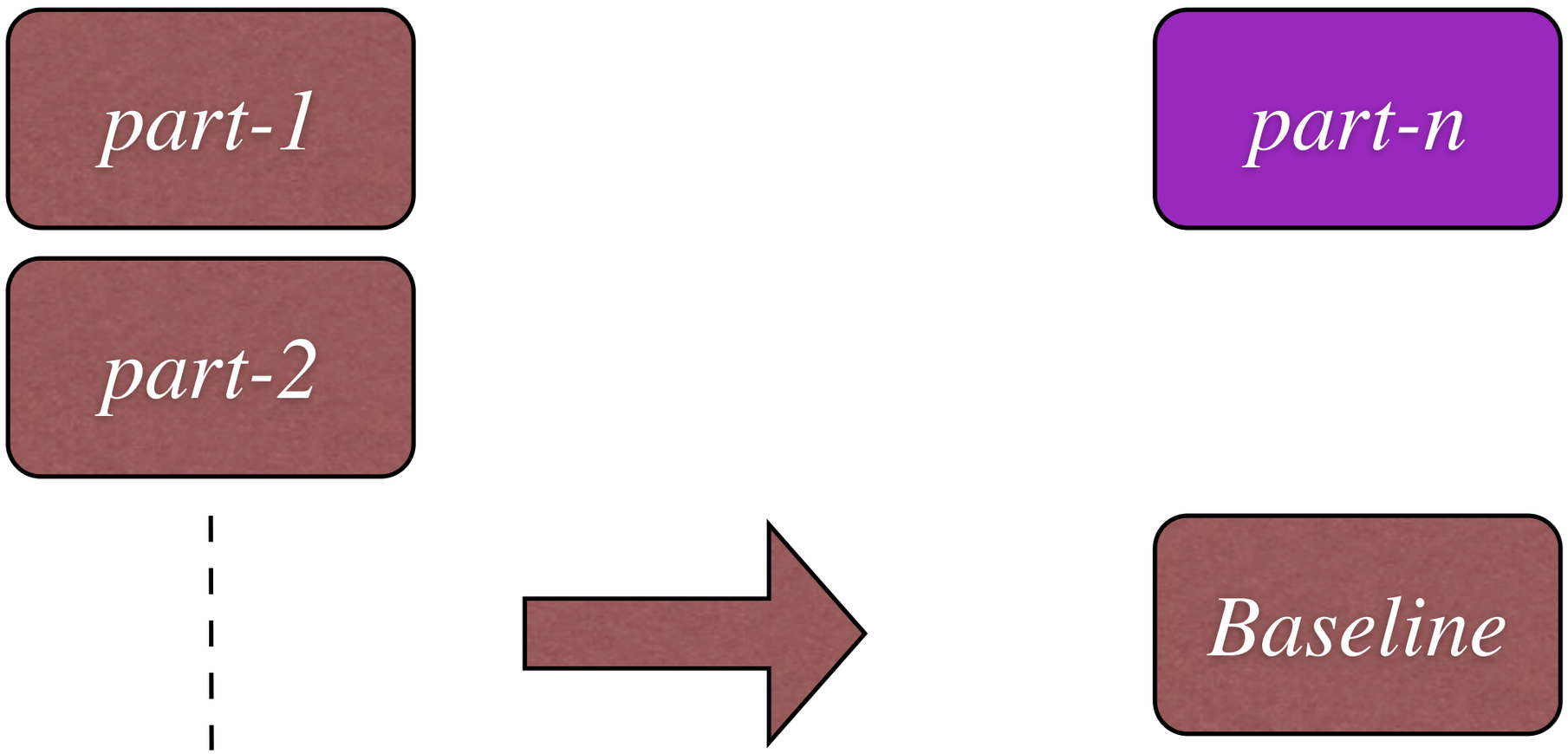
Leave-one-out Baseline Training



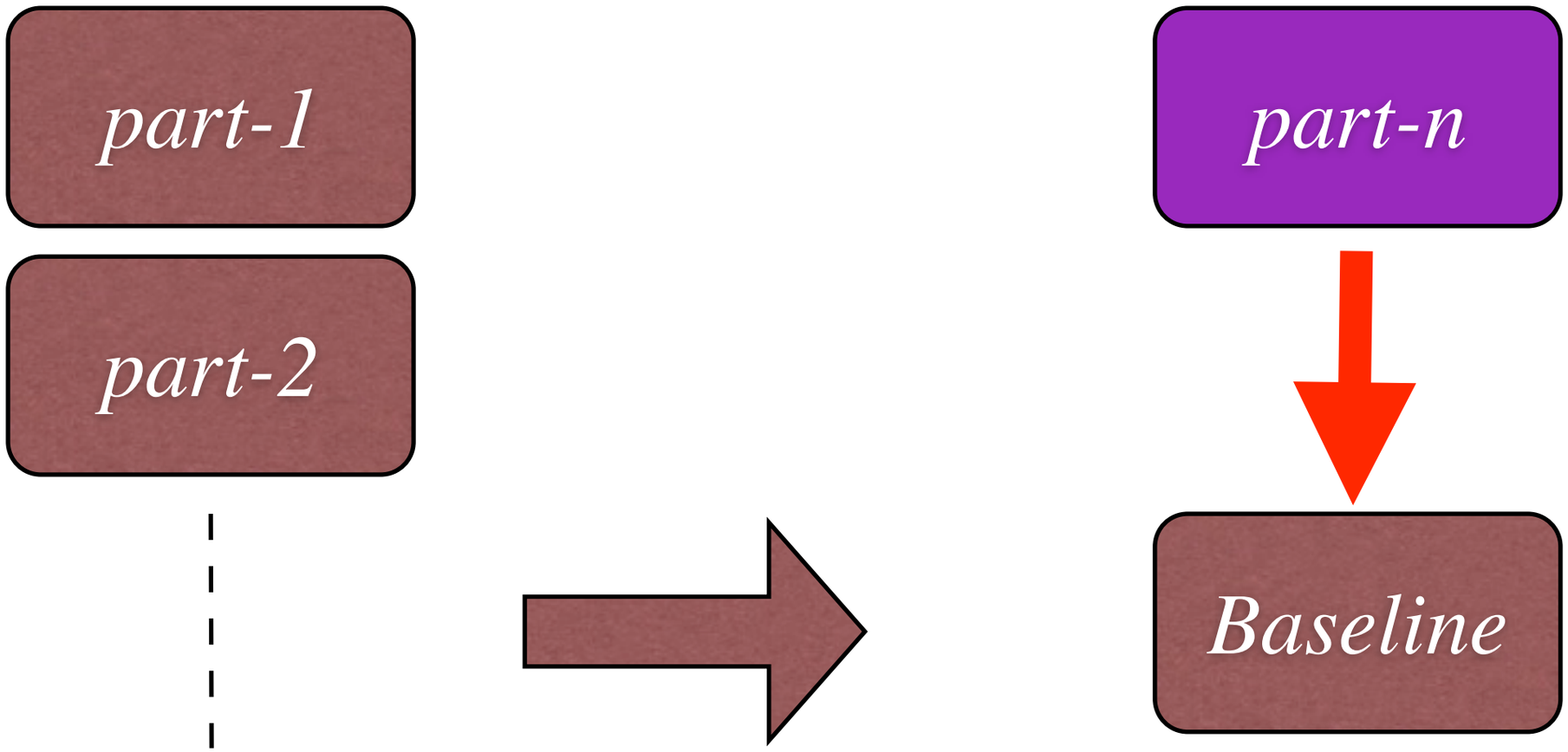
Leave-one-out Baseline Training



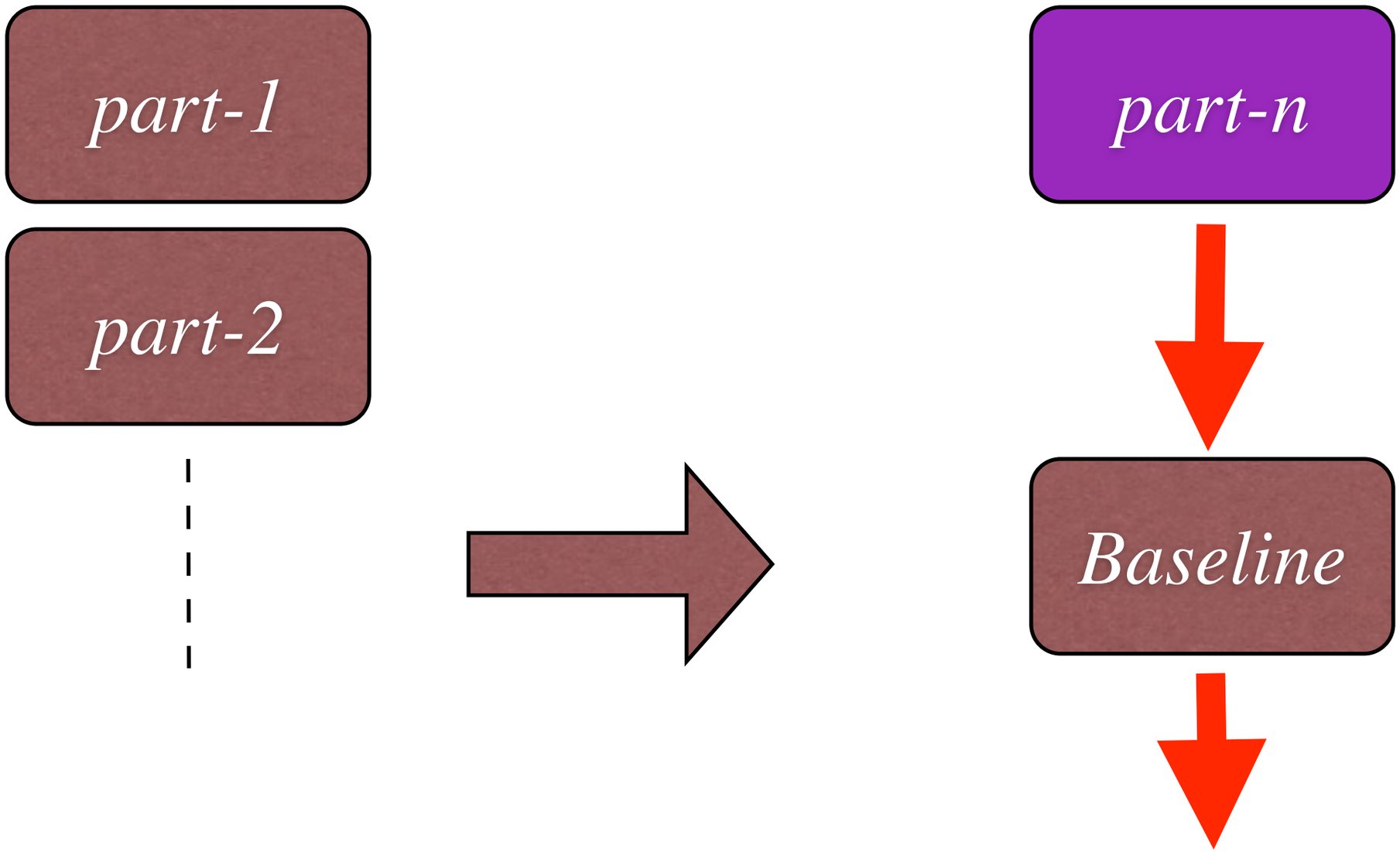
Leave-one-out Baseline Training



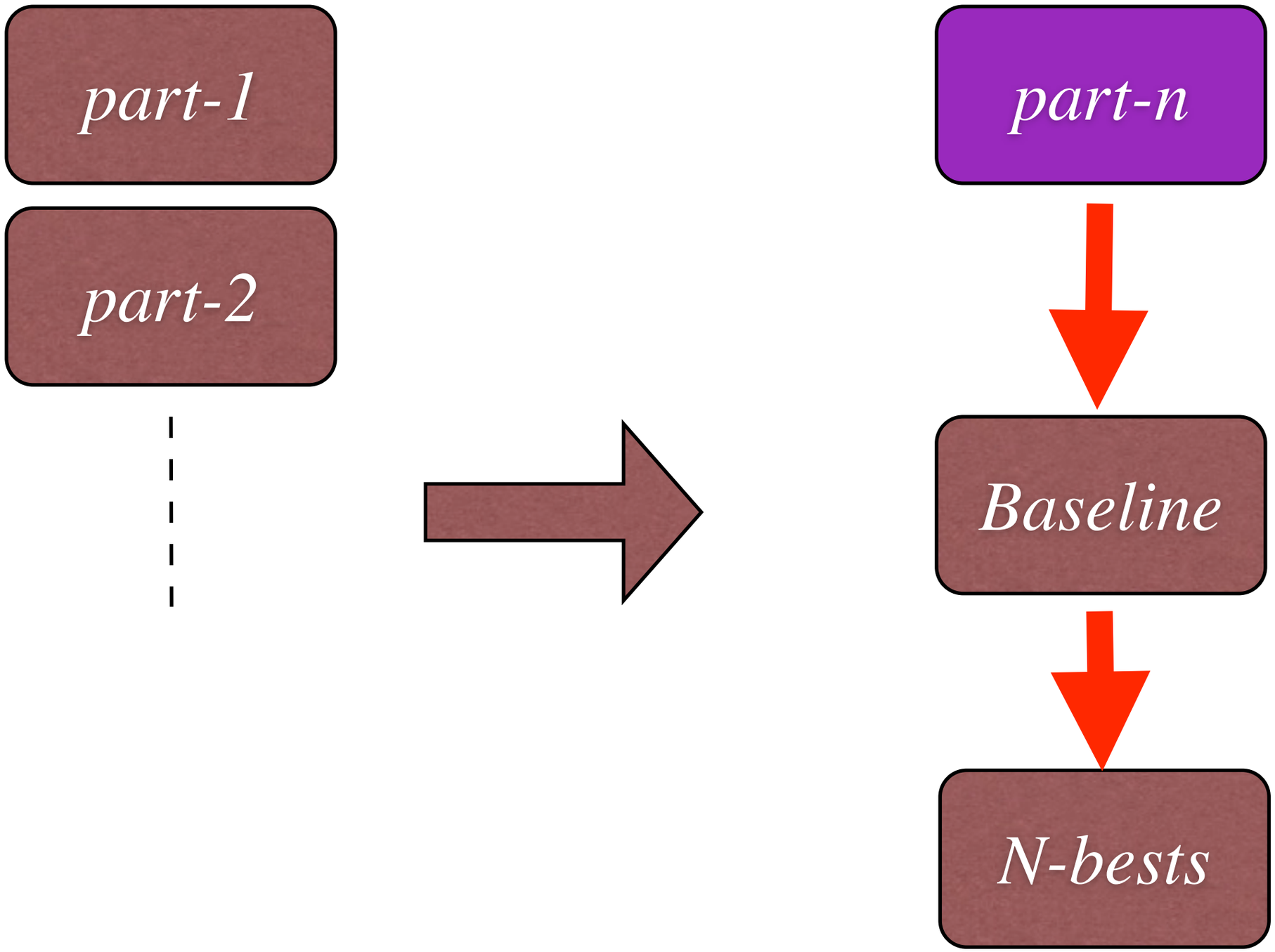
Leave-one-out Baseline Training



Leave-one-out Baseline Training



Leave-one-out Baseline Training



Data Selection

Data Selection

- **Data is very noisy in our MT application**

Data Selection

- **Data is very noisy in our MT application**
  - Human annotation is noisy

Data Selection

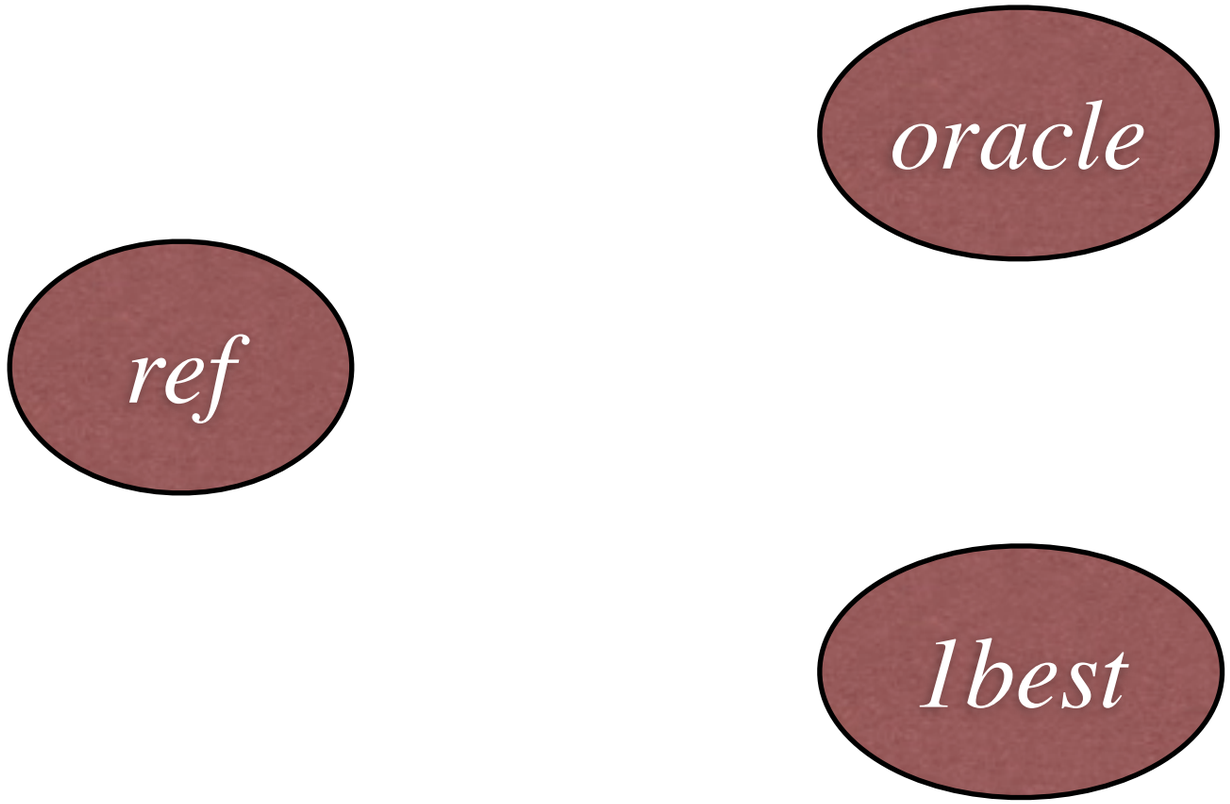
- **Data is very noisy in our MT application**
  - Human annotation is noisy
  - Automatic sentence alignment is noisy

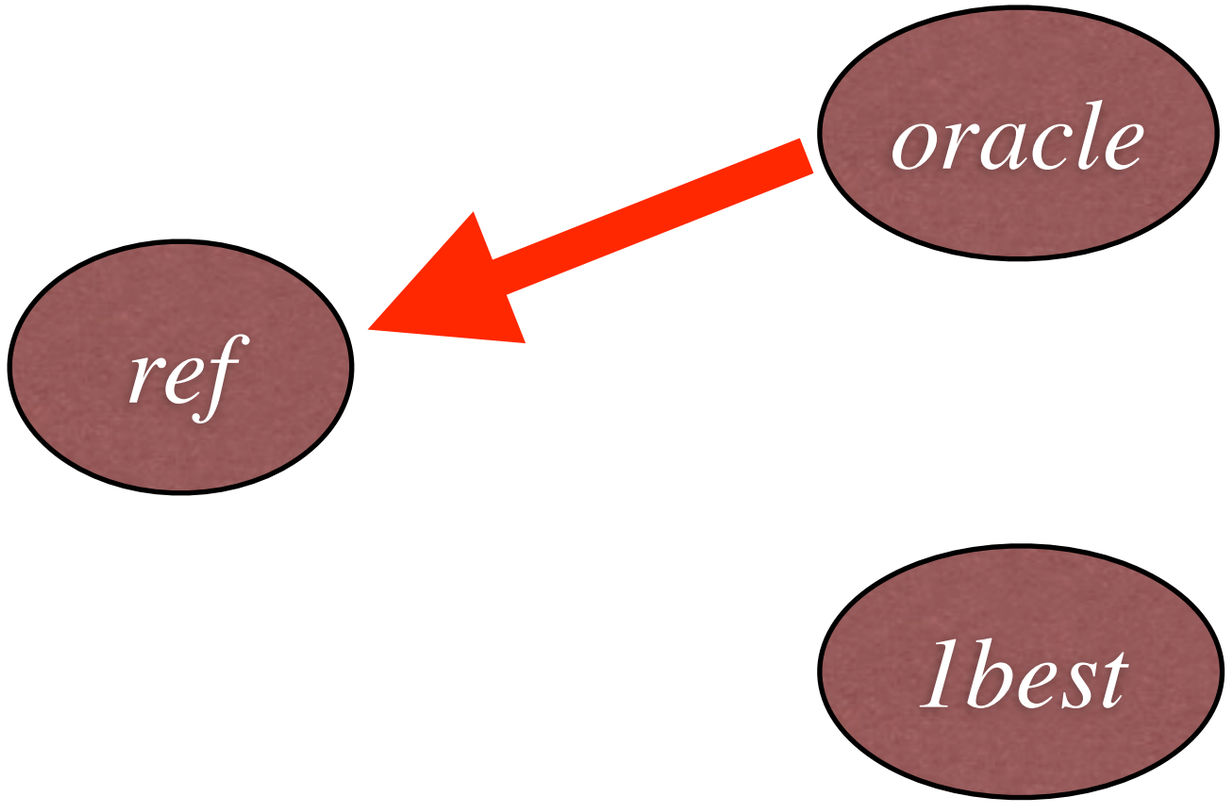
Data Selection

- **Data is very noisy in our MT application**
  - Human annotation is noisy
  - Automatic sentence alignment is noisy
- **We aim to select high-quality training data for discriminative training**

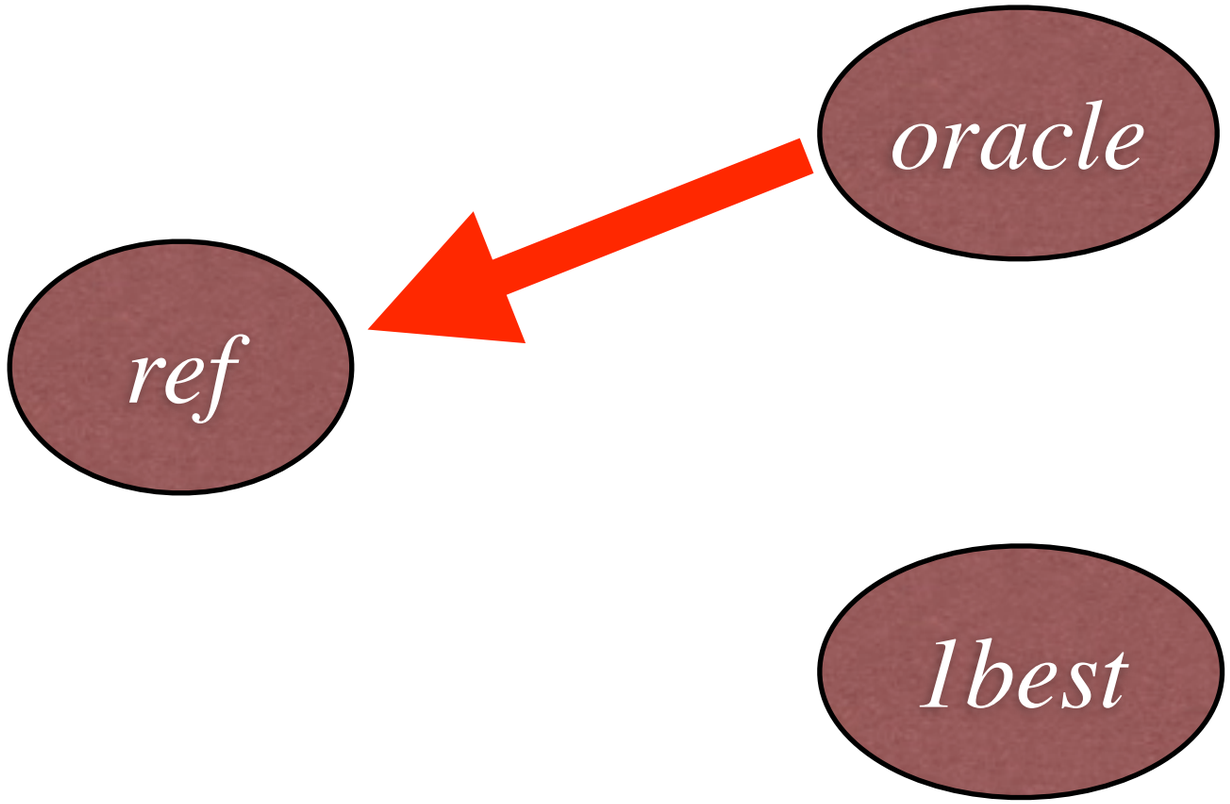
Data Selection

- **Data is very noisy in our MT application**
  - Human annotation is noisy
  - Automatic sentence alignment is noisy
- **We aim to select high-quality training data for discriminative training**
  - An training example will be selected only if it satisfies certain conditions

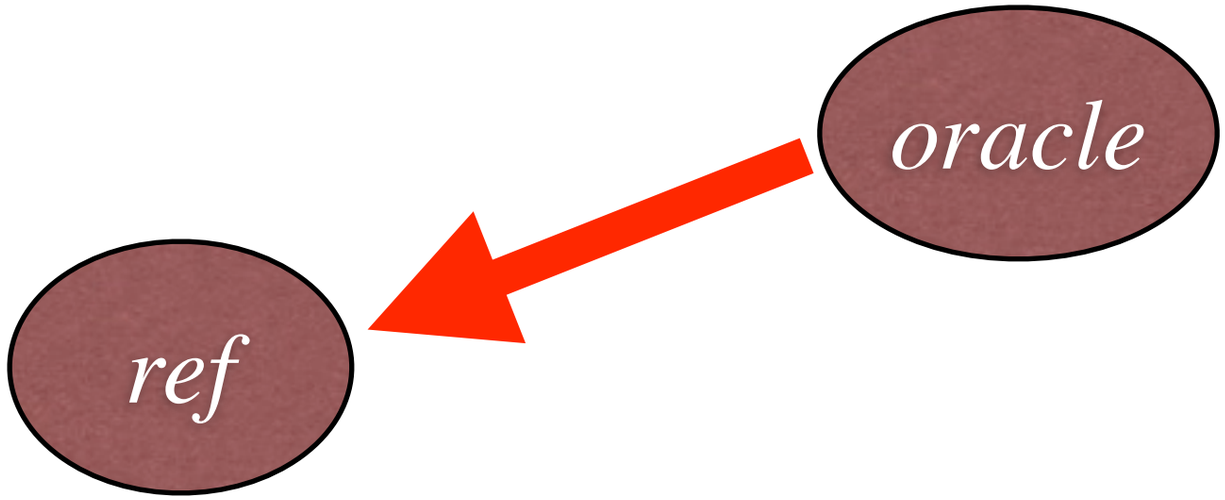

oracle
ref
1best



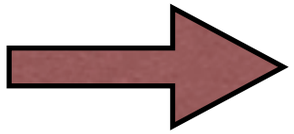
$$\mathbf{G(ref, oracle) > T_1}$$



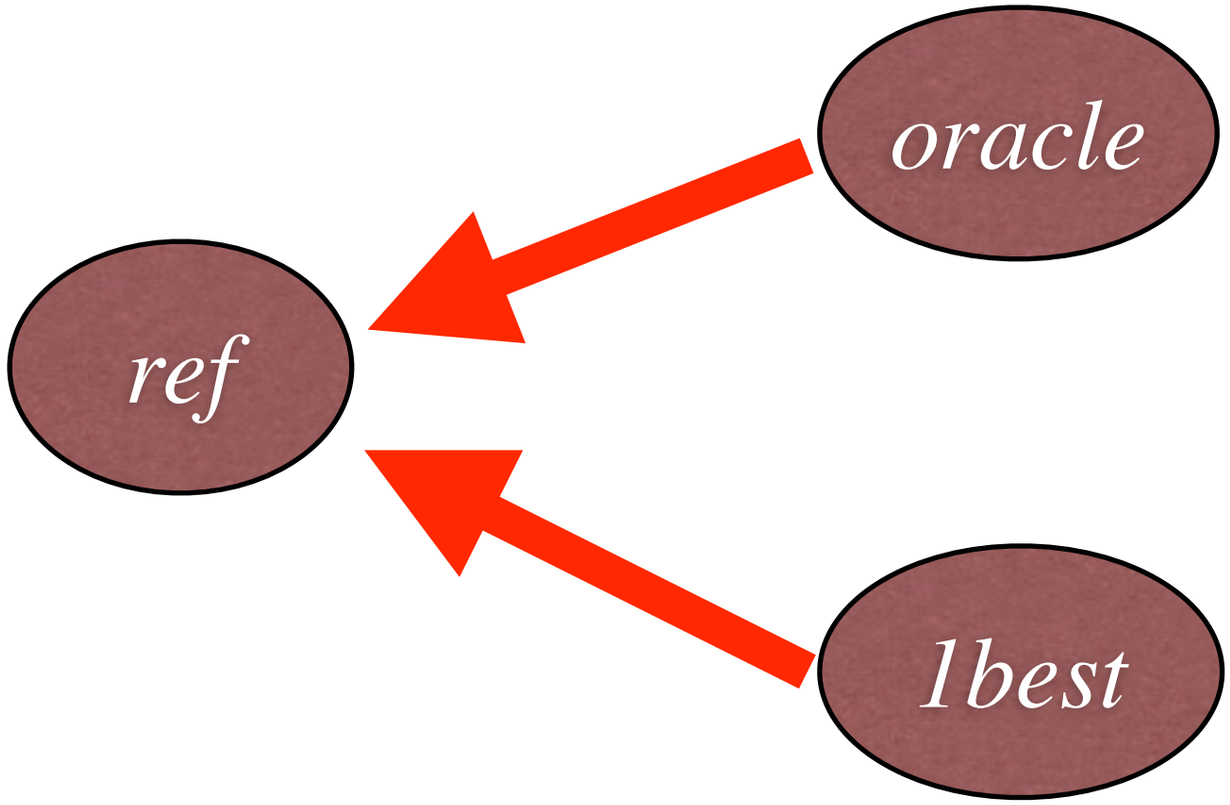
$$\mathbf{G}(\mathbf{ref}, \mathbf{oracle}) > \mathbf{T_1}$$



$$\mathbf{G}(\mathbf{ref}, \mathbf{oracle}) > \mathbf{T_1}$$

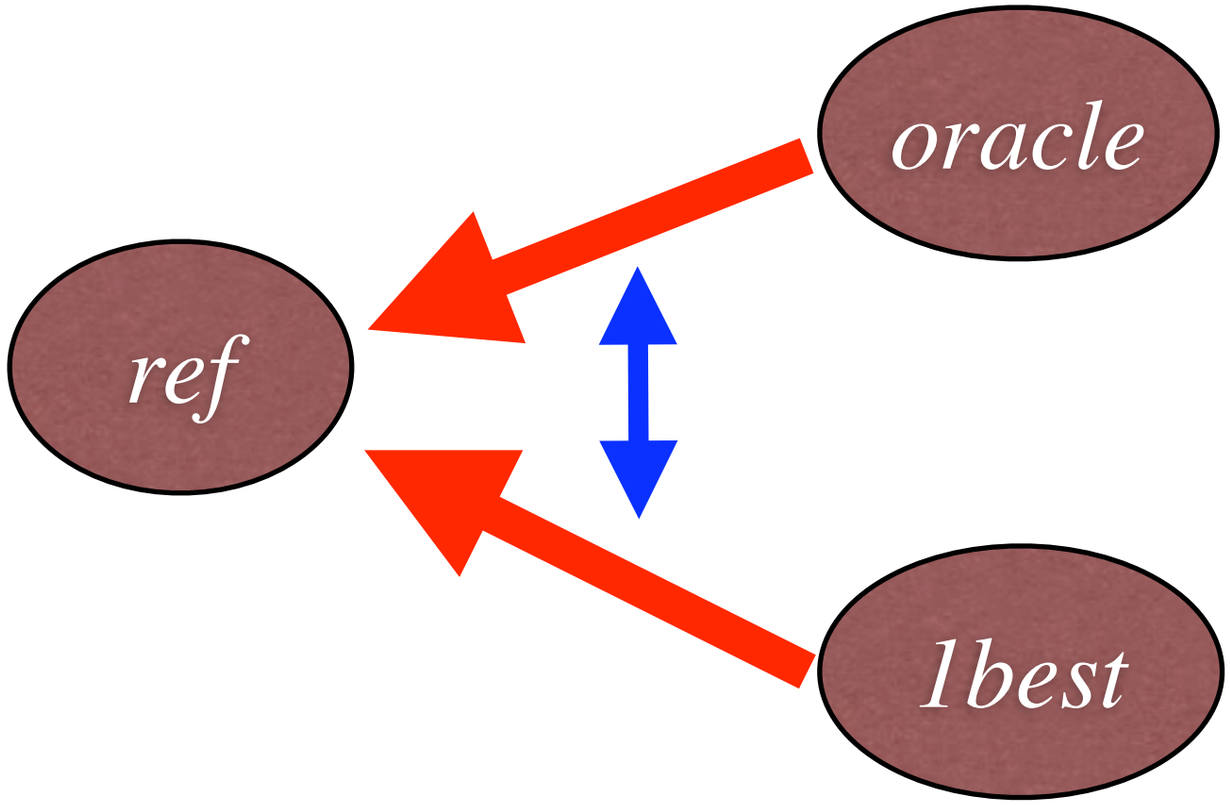


matched translation



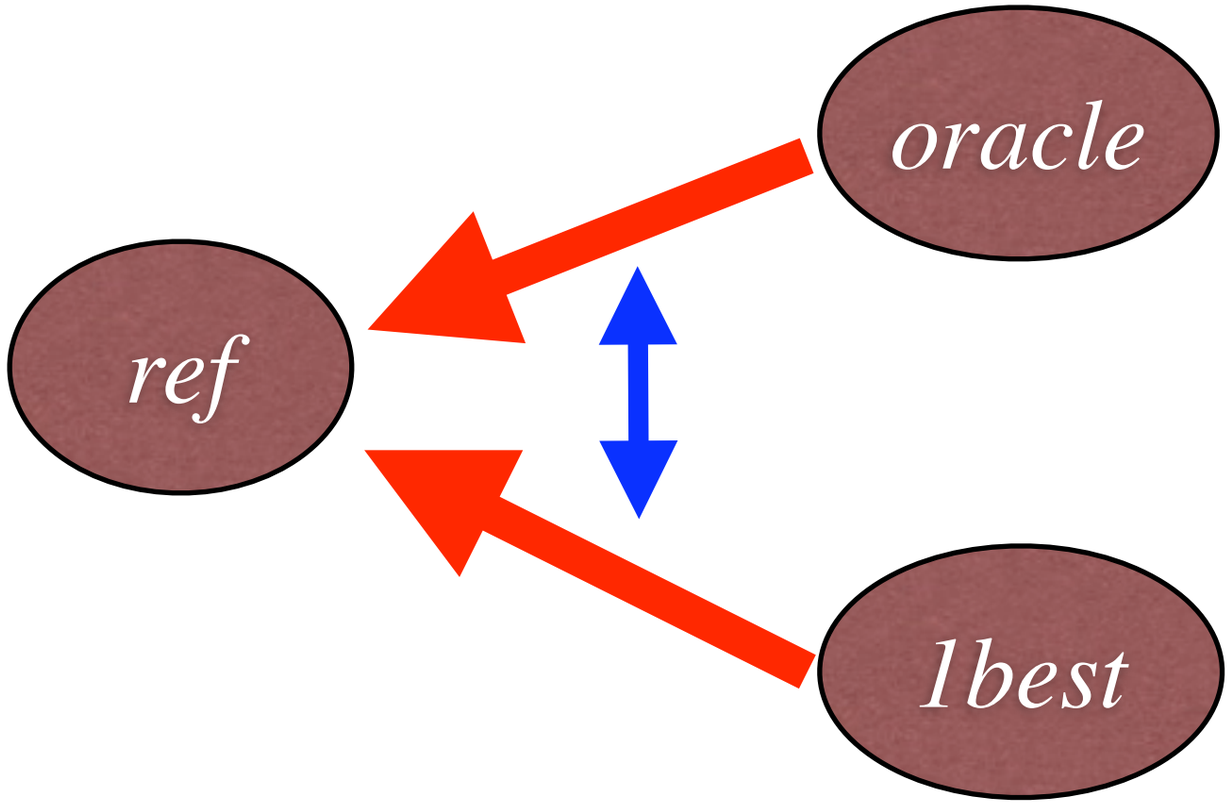
$$\mathbf{G(ref, oracle) > T_1}$$

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$



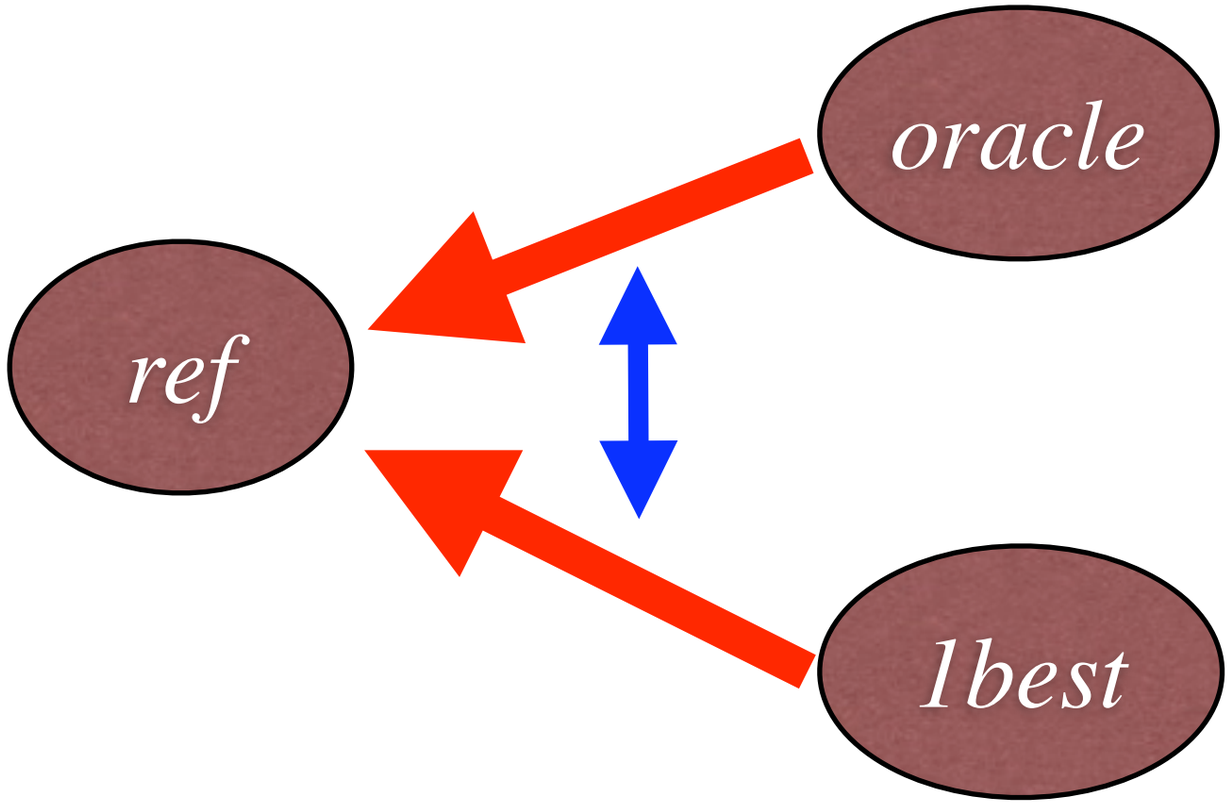
$$\mathbf{G(ref, oracle) > T_1}$$

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$



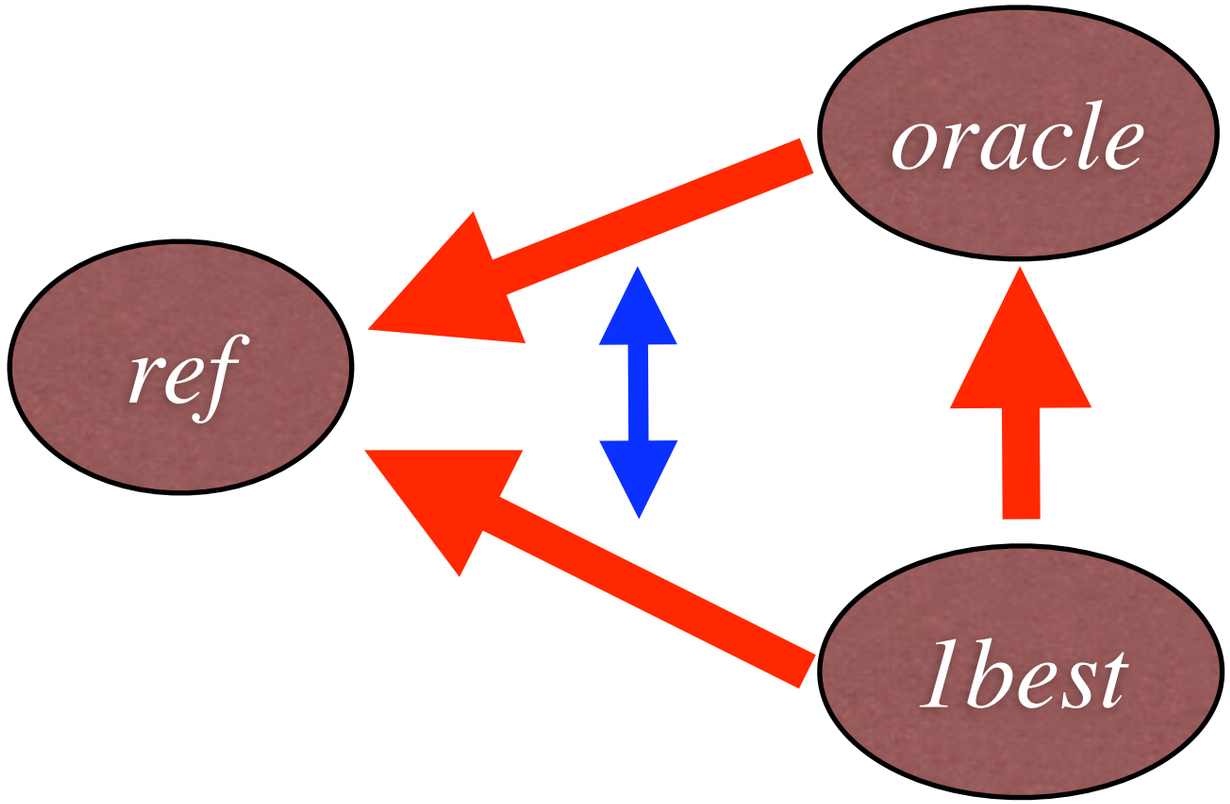
$$\mathbf{G(ref, oracle) > T_1}$$ → *matched translation*

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$ →



$$\mathbf{G(ref, oracle) > T_1}$$ → *matched translation*

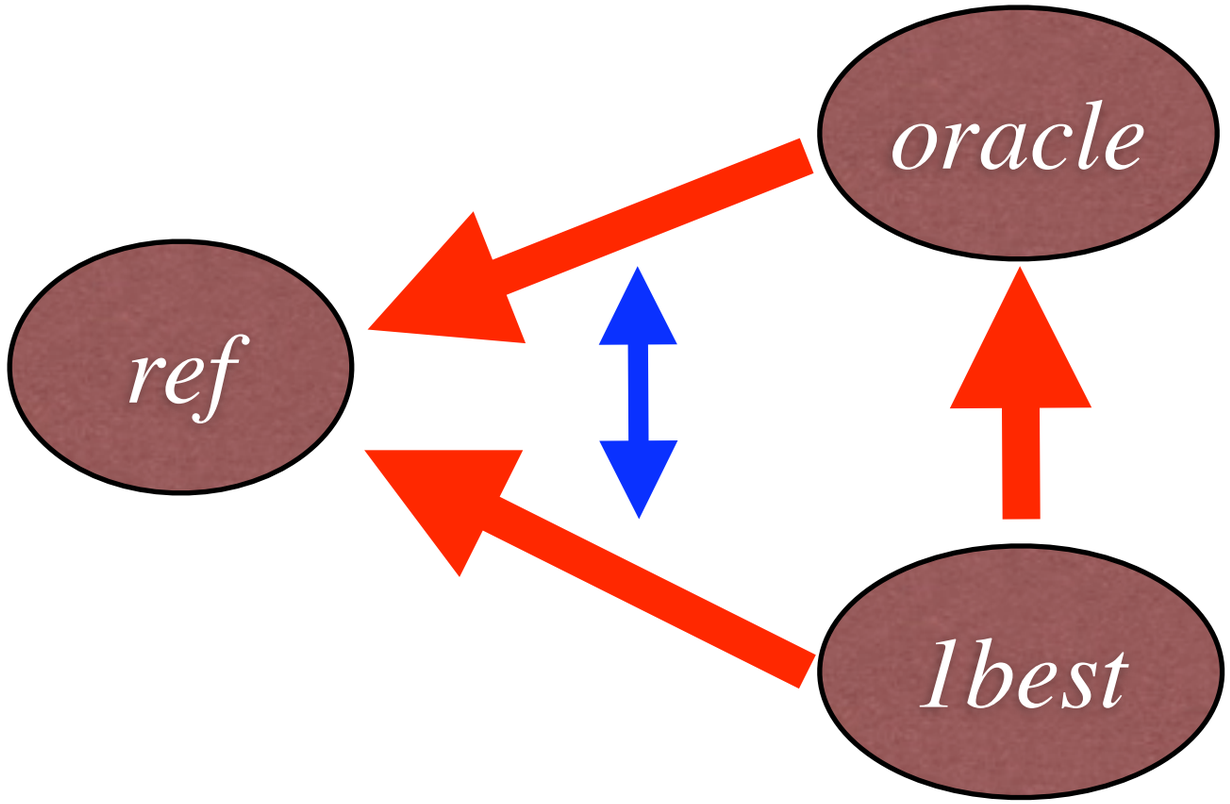
$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$ → *profitable*



$$\mathbf{G(ref, oracle) > T_1}$$ → *matched translation*

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$ → *profitable*

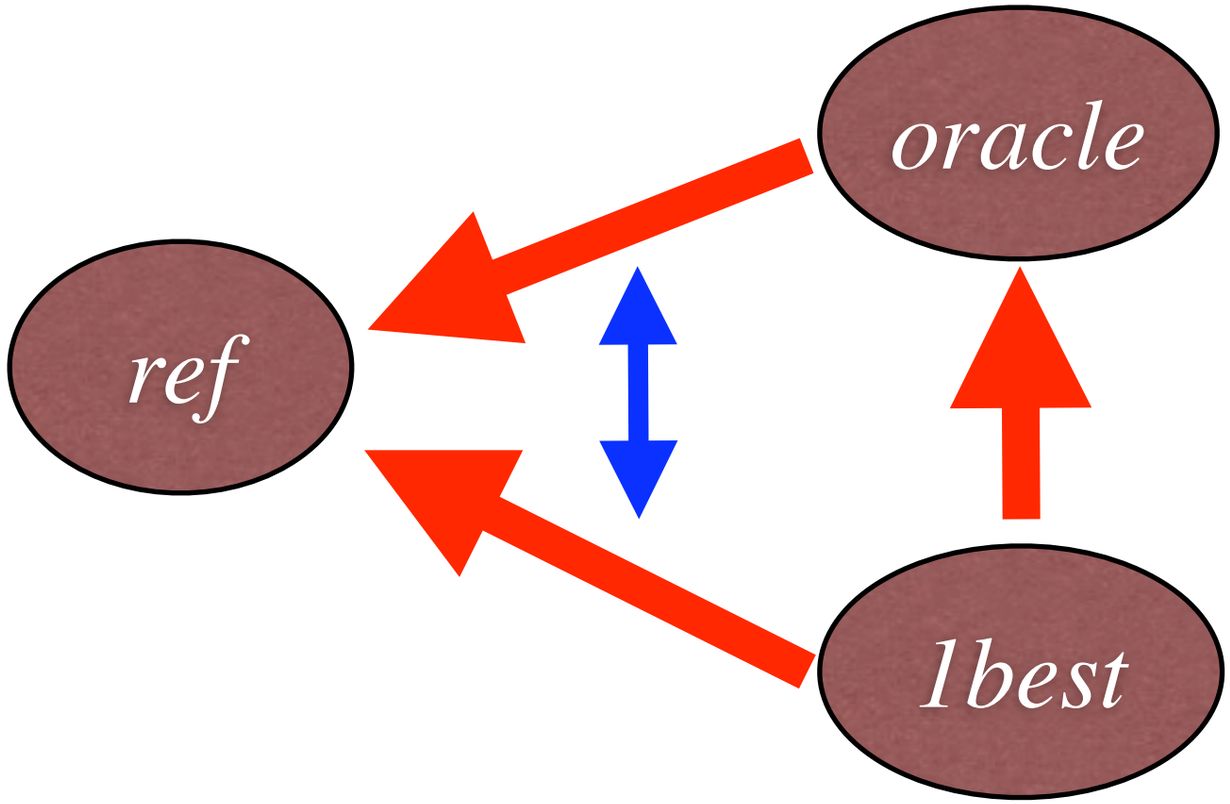
$$\mathbf{G(oracle, 1best) > T_3}$$



$$\mathbf{G(ref, oracle) > T_1}$$ → *matched translation*

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$ → *profitable*

$$\mathbf{G(oracle, 1best) > T_3}$$ →



$$\mathbf{G(ref, oracle) > T_1}$$ → *matched translation*

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$ → *profitable*

$$\mathbf{G(oracle, 1best) > T_3}$$ → *correctable*

Experiments: facts

Language pair	*Chinese to English*
Translation system	*Hiero*
Language model data	*160 M words*
Translation model data	*30M words*
Number of partitions	*30*
DEV set for baseline MERT	*MT03*
DEV set for reranking	*MT04*
Test sets	*MT05, MT06*
N-best size	*300 unique*
Training algorithm	*averaged perceptron*

Data Selection: varying T_1

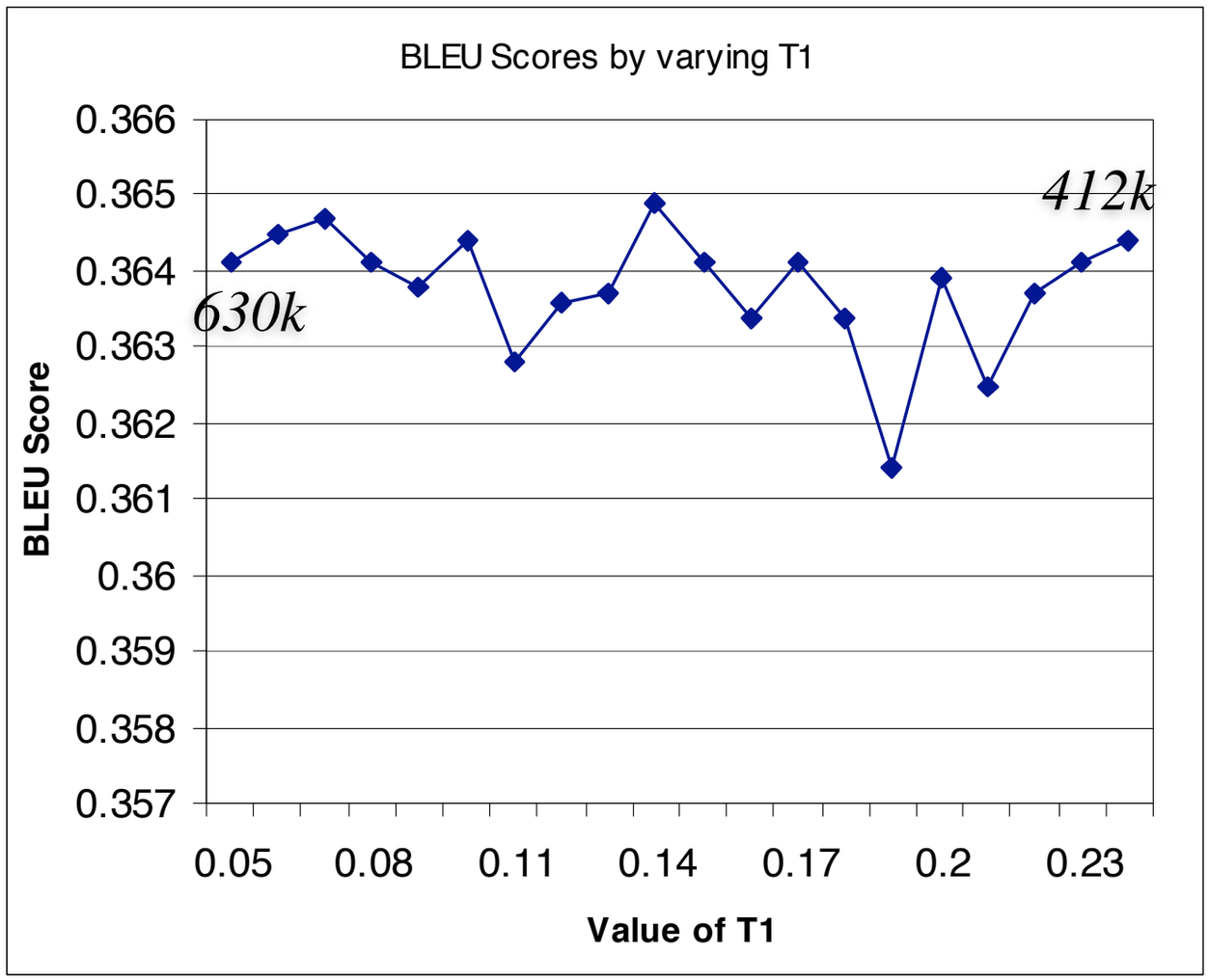


Figure 5: BLEU Scores on MT'04 when varying the value of $T_1 \in [0.05, 0.25]$ with a step size 0.01.

$\mathbf{G(ref, oracle) > T_1}$ → *matched translation*

Data Selection: varying T_2

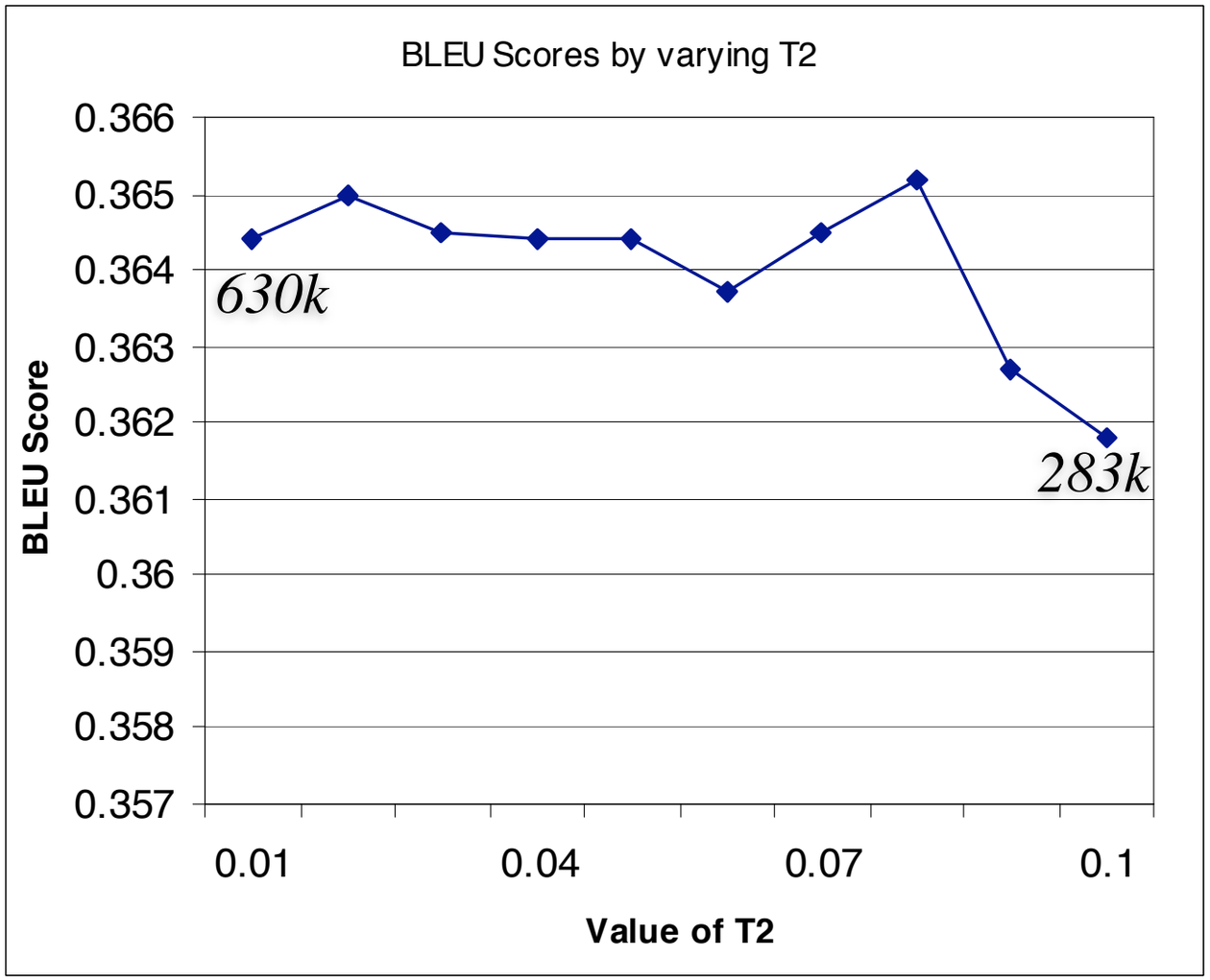


Figure 6: BLEU Scores on MT'04 when varying the value of $T_2 \in [0.01, 0.10]$ with a step size 0.01.

$$\mathbf{G(ref, oracle) - G(ref, 1best) > T_2}$$ → *profitable*

Data Selection: varying T_3

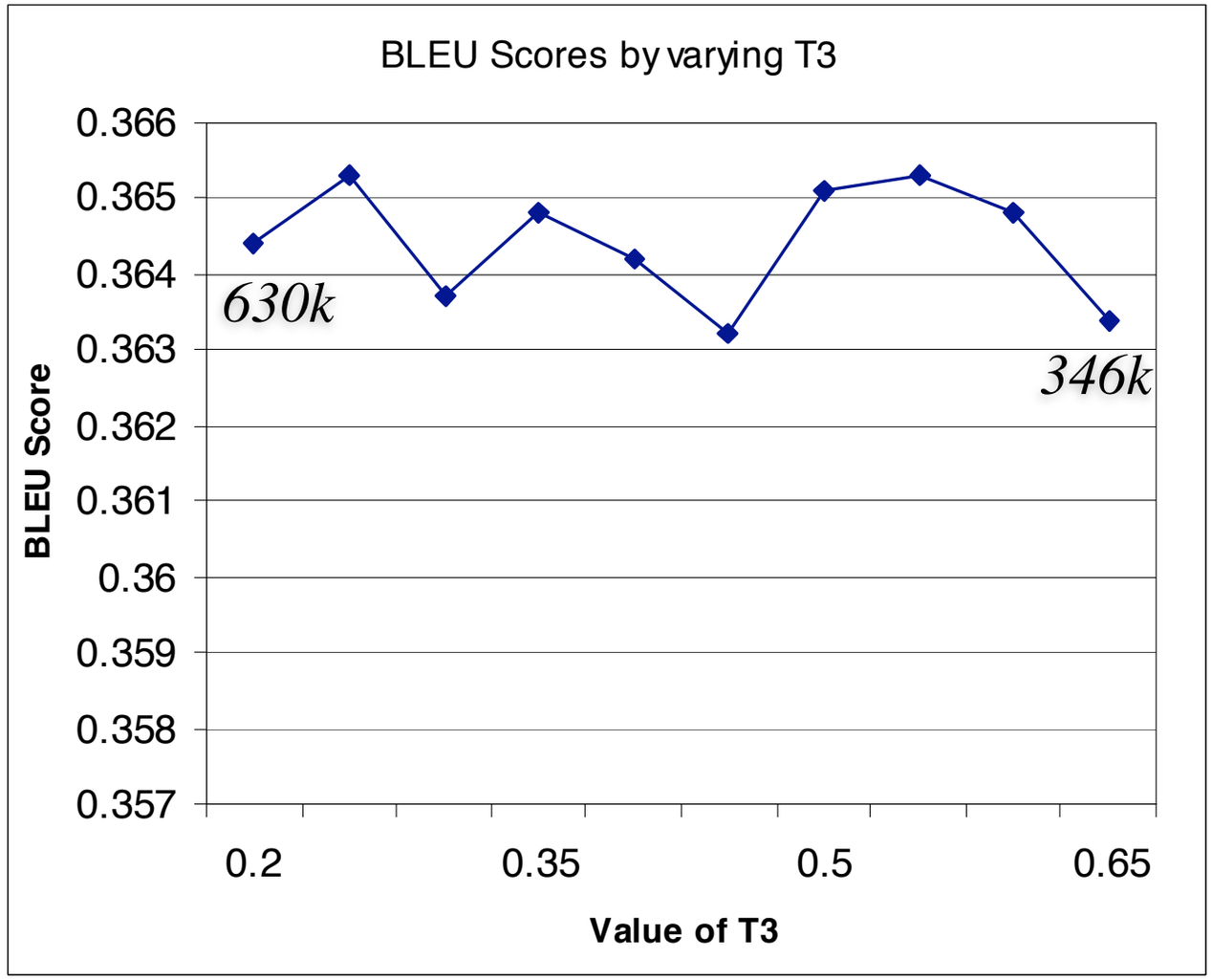


Figure 7: BLEU Scores on MT'04 when varying the value of $T_3 \in [0.20, 0.75]$ with a step size 0.05.

$$\mathbf{G(oracle, 1best) > T_3} \Rightarrow \textit{correctable}$$

Experiments: reranking results

Task	Baseline	Reranking	
		Full	Selected
MT'04	0.357	**0.365**	0.365
MT'05	0.326	0.332	**0.333**
MT'06	0.283	0.292	**0.294**

Experiments: reranking results

Task	Baseline	Reranking	
		Full	Selected
MT'04	0.357	**0.365**	0.365
MT'05	0.326	0.332	**0.333**
MT'06	0.283	0.292	**0.294**

T_1	T_2	T_3	Selected data
0.10	0.01	0.25	610K out of 1M

Experiments: reranking results

Task	Baseline	Reranking	
		Full	Selected
MT'04	0.357	**0.365**	0.365
MT'05	0.326	0.332	**0.333**
MT'06	0.283	0.292	**0.294**

n-gram	active
1-gram	34k
2-gram	1908k

Summary

Summary

- **We have developed a discriminative n-gram LM to rerank MT outputs**

Summary

- We have developed a discriminative n-gram LM to rerank MT outputs
- Discriminative LM reranking improves the translation quality over a state of the art system

Summary

- We have developed a discriminative n-gram LM to rerank MT outputs
- Discriminative LM reranking improves the translation quality over a state of the art system
- With data selection, we can train a better/comparable model using less data

Hypergraph-based Discriminative Rescoring

Hypergraph-based Discriminative Rescoring

- generate a hypergraph (instead of an n-best) for each Chinese sentence

Hypergraph-based Discriminative Rescoring

- generate a hypergraph (instead of an n-best) for each Chinese sentence
- identify oracle translations on the hypergraph

Hypergraph-based Discriminative Rescoring

- generate a hypergraph (instead of an n-best) for each Chinese sentence
- identify oracle translations on the hypergraph
- train a model and use it in decoding on a hypergraph

Hypergraph-based Discriminative Rescoring

- generate a hypergraph (instead of an n-best) for each Chinese sentence
- identify oracle translations on the hypergraph
- train a model and use it in decoding on a hypergraph
- the hypergraph is pruned using the posterior pruning

Hypergraph rescoring results

System		MT04	MT05	MT06
Baseline	Chiang'07	34.6	31.8	NA
	Ours	35.7	32.6	28.3
N-best		**36.5**	**33.3**	**29.4**
Hypergraph		35.9	33.0	28.2

Joshua: an open-source parsing-based MT decoder

- Team members
  - **JHU**: Zhifei Li, Chris Callison-Burch, Sanjeev Khudanpur, Wren Thornton, Jonathan Weese, and Omar Zaidan
  - **UMD**: Chris Dyer
  - **U of Minnesota**: Lane Schwartz
- Functions
  - Chart-parsing, pruning, language model integration, kbest extraction, distributed and parallel decoding
  - Suffix-array based grammar extraction
  - Minimum error rate training

Thank you!
谢谢!