Unsupervised Ensemble of Ranking Models for News Comments Using Pseudo Answers

Soichiro Fujita 1, Hayato Kobayashi 2, Manabu Okumura 1

1. Tokyo Institute of Technology

2. Yahoo Japan Corporation / RIKEN AIP

Background

Task: Ranking comments on online news services

Goal: Display high quality comments

Problem: `` high quality'' has complex factors

News Article   User comments   Ranking

Ranking news comments is difficult

- We have <u>various situations</u> of judging whether a comment is good
    - Indicating rare user experiences
    - Providing new ideas
    - Causing discussions

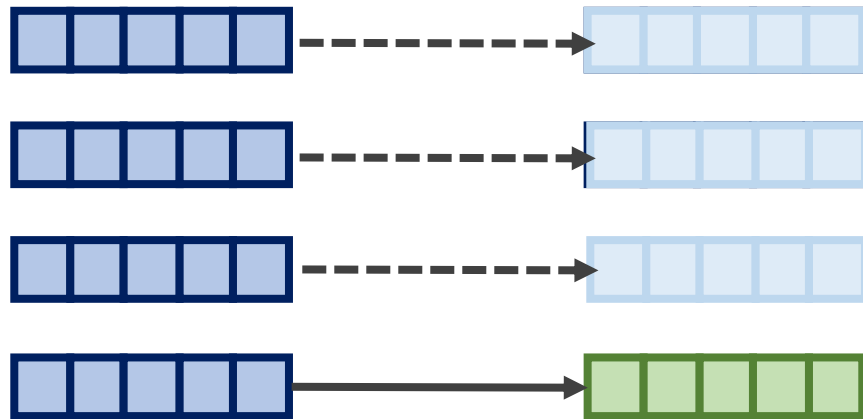- Ranking models often <u>fail to capture these information</u>

How to deal with this problem? → Ensemble techniques

If we prepare many models,
some models can capture these information

Two Basic Ensemble Techniques

Selecting

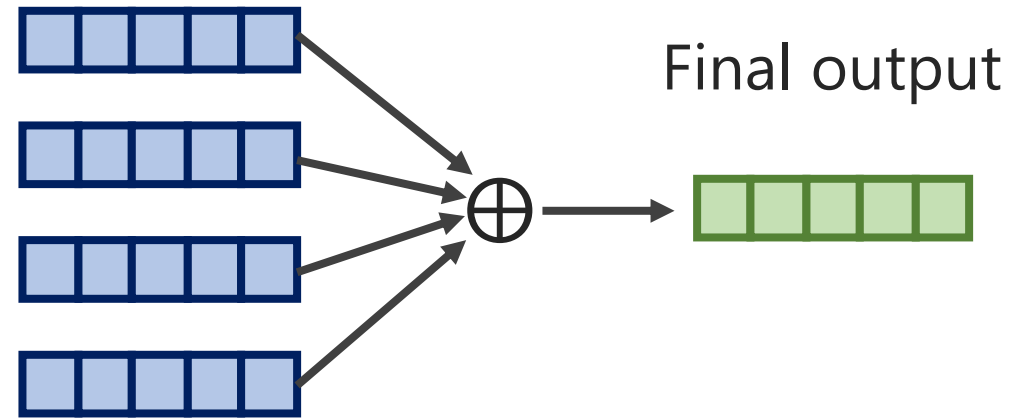model outputs   selected outputs

✓ Denoising lower accuracy model
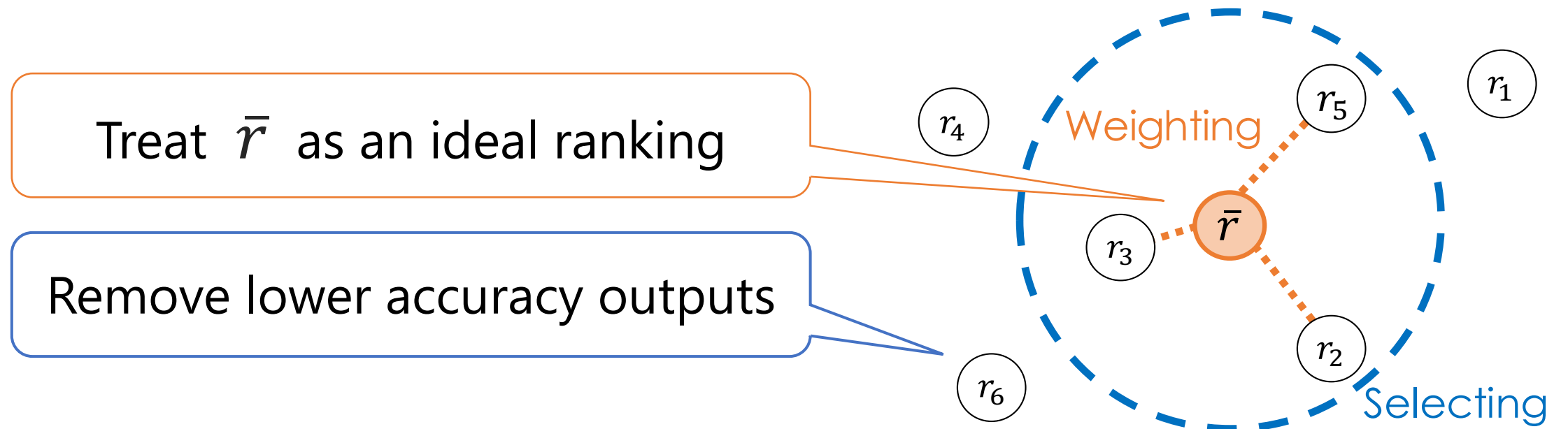
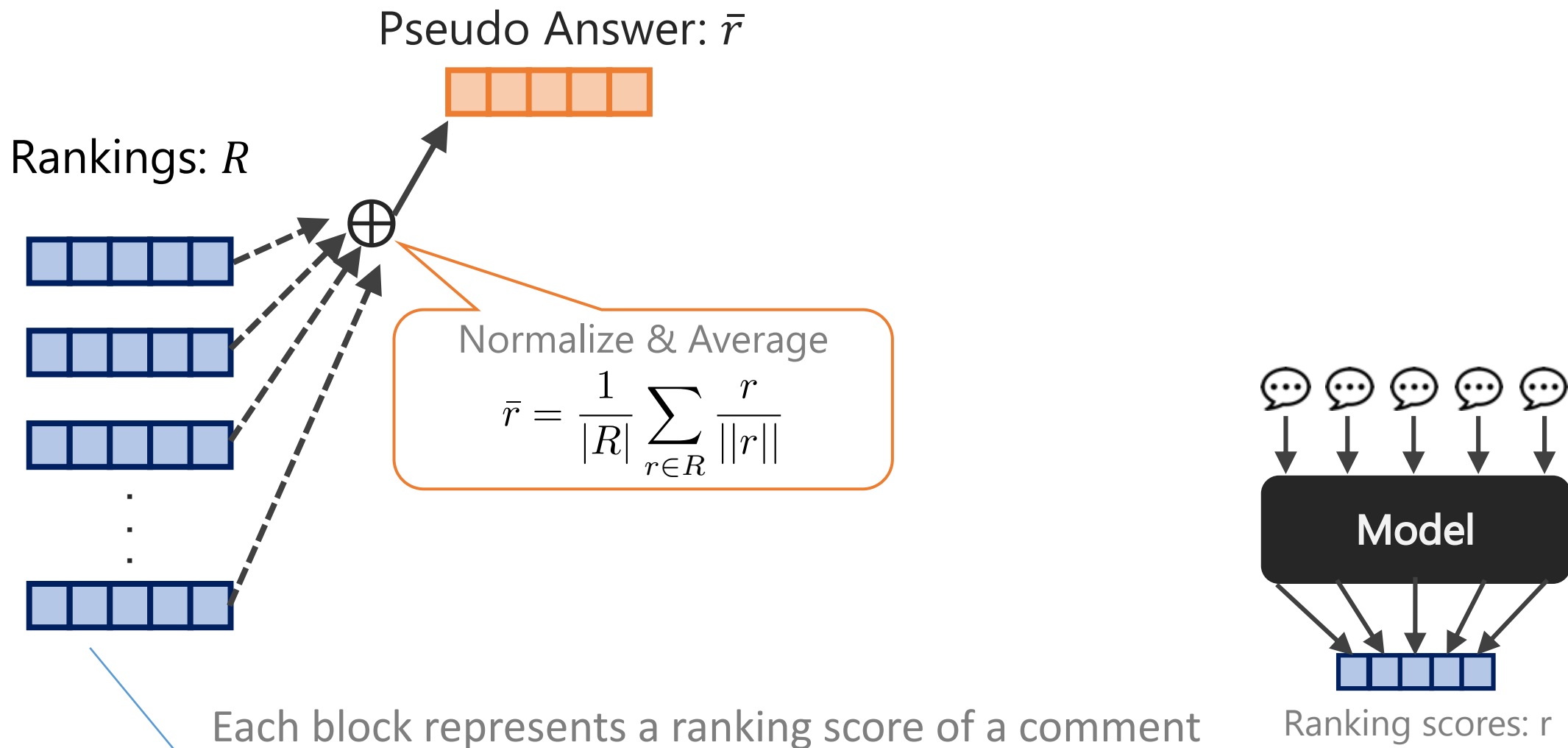✗ Depend on a single model output

Averaging

model outputs

Final output

✓ Make up for other models' mistakes

✗ Lower accuracy model could be noise

Proposed method: HPA ensemble

- **H**ybrid method using the **P**seudo **A**nswer

  - Hybrid of an output selection and a typical averaging method

  - Dynamic denoising of outputs via a pseudo answer $\bar{r}$

Treat $\bar{r}$ as an ideal ranking

Remove lower accuracy outputs

r_4

r_5

r_1

Weighting

r_3

$\bar{r}$

r_6

r_2

Selecting

Step1: Calculate a pseudo answer

Pseudo Answer: $\bar{r}$

Rankings: R

Normalize & Average

$$\bar{r} = \frac{1}{|R|} \sum_{r \in R} \frac{r}{||r||}$$

Model

Each block represents a ranking score of a comment

Ranking scores: r

Step2: Calculate similarity scores of each predicted ranking

Pseudo Answer: $\bar{r}$

Rankings: R

Regard the pseudo answer as an ideal ranking

$\oplus$

0.87

0.75

0.65

.
.
.

0.92

Using NDCG
for a similarity function

$$g(r) = \text{NDCG}(\bar{r}, r)$$
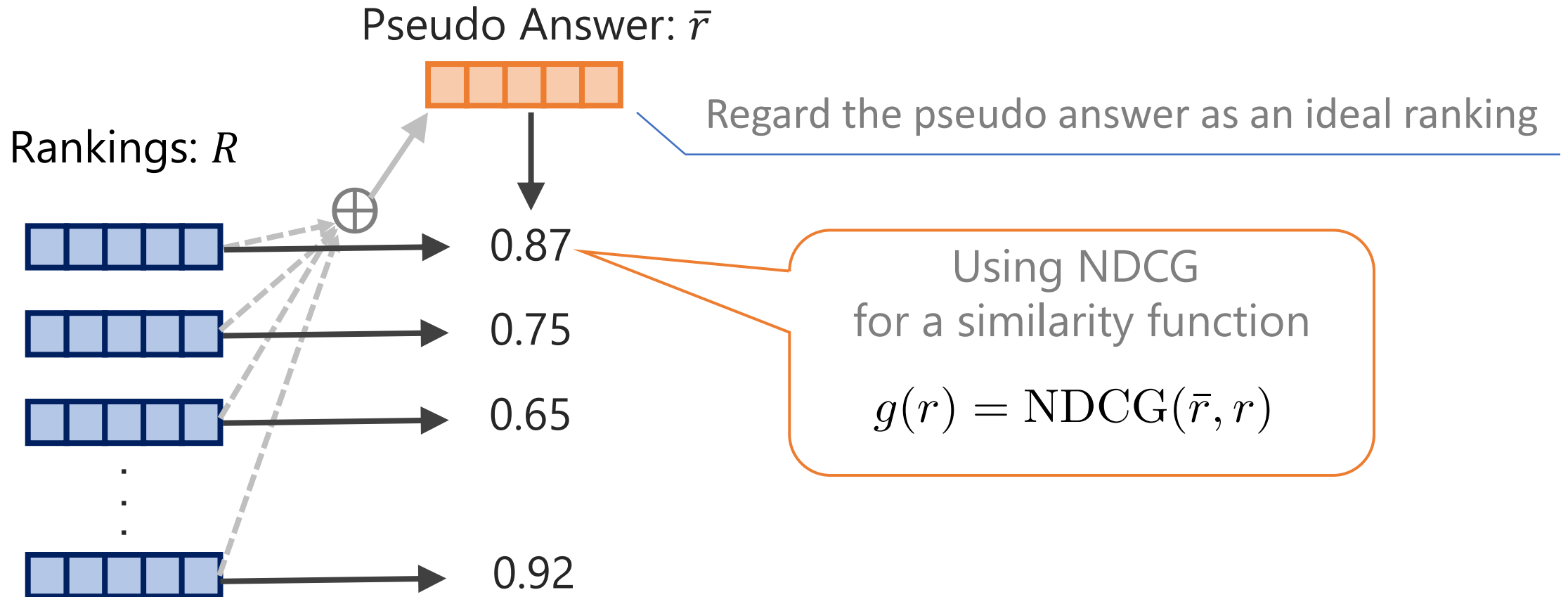
Step3: Calculate the final ranking from similarity scores
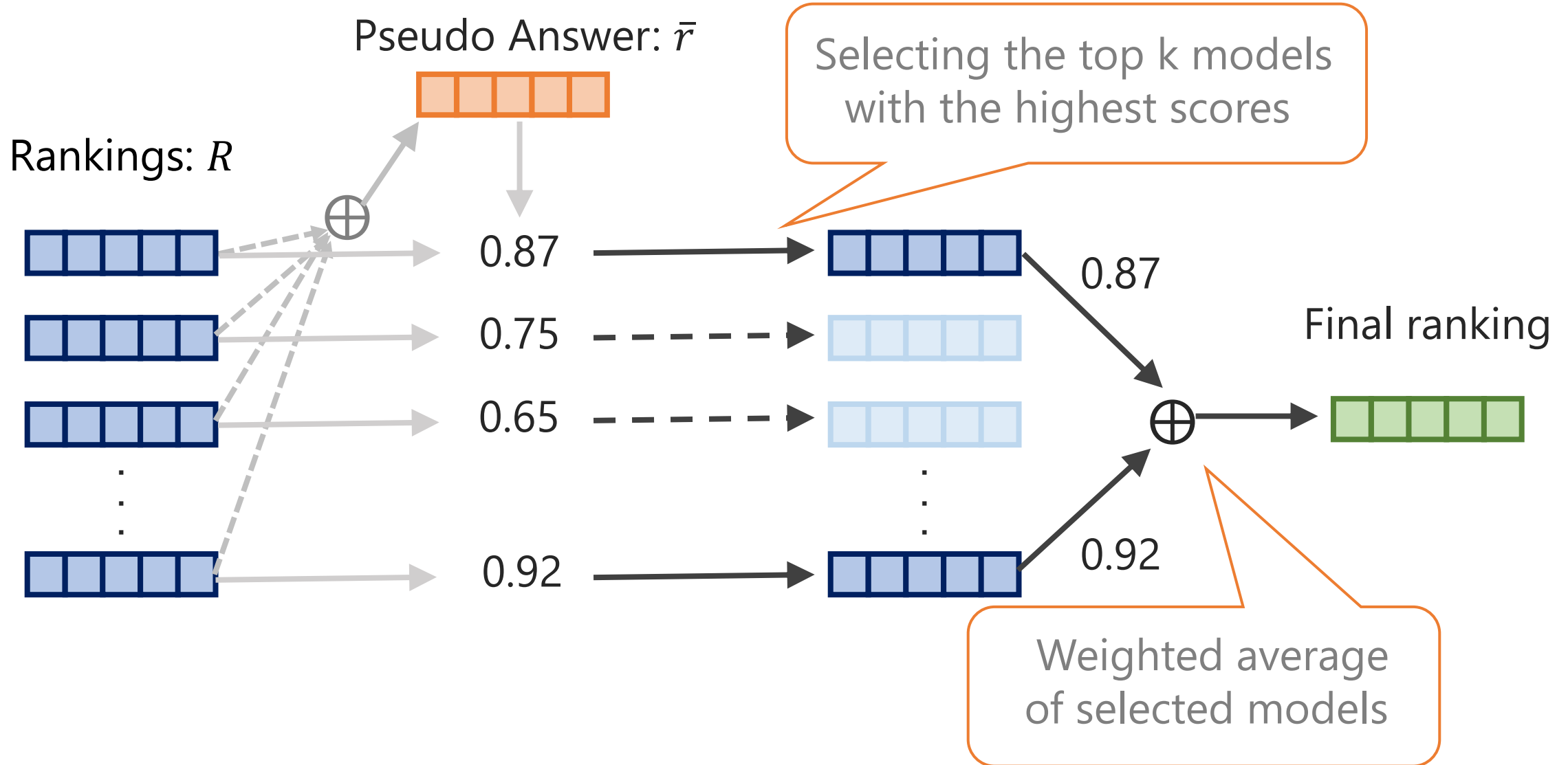
Experimental Settings

Dataset: YJ Constructive Comment Ranking Dataset

Train $1{,}300$ articles, Validation 113 articles, Test 200 articles

(each article associated with more than 100 comments)

Models: LSTM-based RankNet

Prepared 100 different models by random initialization

Metrics: NDCG@k and Precision@k ($k \in \{1, 5, 10\}$)

Evaluation Results

	Methods	NDCG			Prec.		
		@1	@5	@10	@1	@5	@10
Best single model	RankNet	76.35	77.97	79.52	15.0	33.20	42.99
Unsupervised baseline	NormAvg	**79.83**	**80.77**	**82.16**	**17.08**	**37.18**	46.48
Supervised baseline	SupWeight	78.64	80.33	81.94	16.28	35.47	**46.58**
Ours	HPA	79.87	81.43	82.33	17.08	37.39	47.34
Ours w/o weighting	SPA	79.68	80.96	82.19	17.08	35.87	46.68
Ours w/o selecting	WPA	79.87	81.39	82.17	17.08	37.88	46.63

This is a part of the results.
Please see Table 1 in our paper if you want to find other baselines.

Evaluation Results

	Methods	NDCG			Prec.		
		@1	@5	@10	@1	@5	@10
Best single model	RankNet	76.35	77.97	79.52	15.0	33.20	42.99
Unsupervised baseline	NormAvg	**79.83**	**80.77**	**82.16**	**17.08**	**37.18**	46.48
Supervised baseline	SupWeight	78.64	80.33	81.94	16.28	35.47	**46.58**
Ours	HPA	79.87	81.43	82.33	17.08	37.39	47.34
Ours w/o weighting	SPA	79.68	80.96	82.19	17.08	35.87	46.68
Ours w/o selecting							

Our method achieved the best performance

Evaluation Results

	Methods	NDCG			Prec.		
		@1	@5	@10	@1	@5	@10
Best single model	RankNet	76.35	77.97	79.52	15.0	33.20	42.99
Unsupervised baseline	NormAvg	**79.83**	**80.77**	**82.16**	**17.08**	**37.18**	46.48
Supervised baseline	SupWeight	78.64	80.33	81.94	16.28	35.47	**46.58**
Ours	HPA	79.87	81.43	82.33	17.08	37.39	47.34
Ours w/o weighting	SPA	79.68	80.96	82.19	17.08	35.87	46.68
Ours w/o selecting	WPA	79.87	81.39	82.17	17.08	37.88	46.63

Hybrid of weighting and selecting is effective

Conclusion

Proposed Method:
- A hybrid unsupervised method using pseudo answers

Result:
- Our method achieved the best performance
- Denoising predicted rankings using the pseudo answer is effective

Future work:
- Combine various types of network structures
- Investigate effectiveness of our methods on other ranking datasets