NAACL 2024

RST-LoRA: A Discourse-Aware Low-Rank Adaptation for Long Document Abstractive Summarization

Dongqi Pu, Vera Demberg

Department of Computer Science
Department of Language Science and Technology
Saarland Informatics Campus, Saarland University, Germany
dongqi.me@gmail.com

UNIVERSITÄT DES SAARLANDES

European Research Council
Established by the European Commission

TL;DR

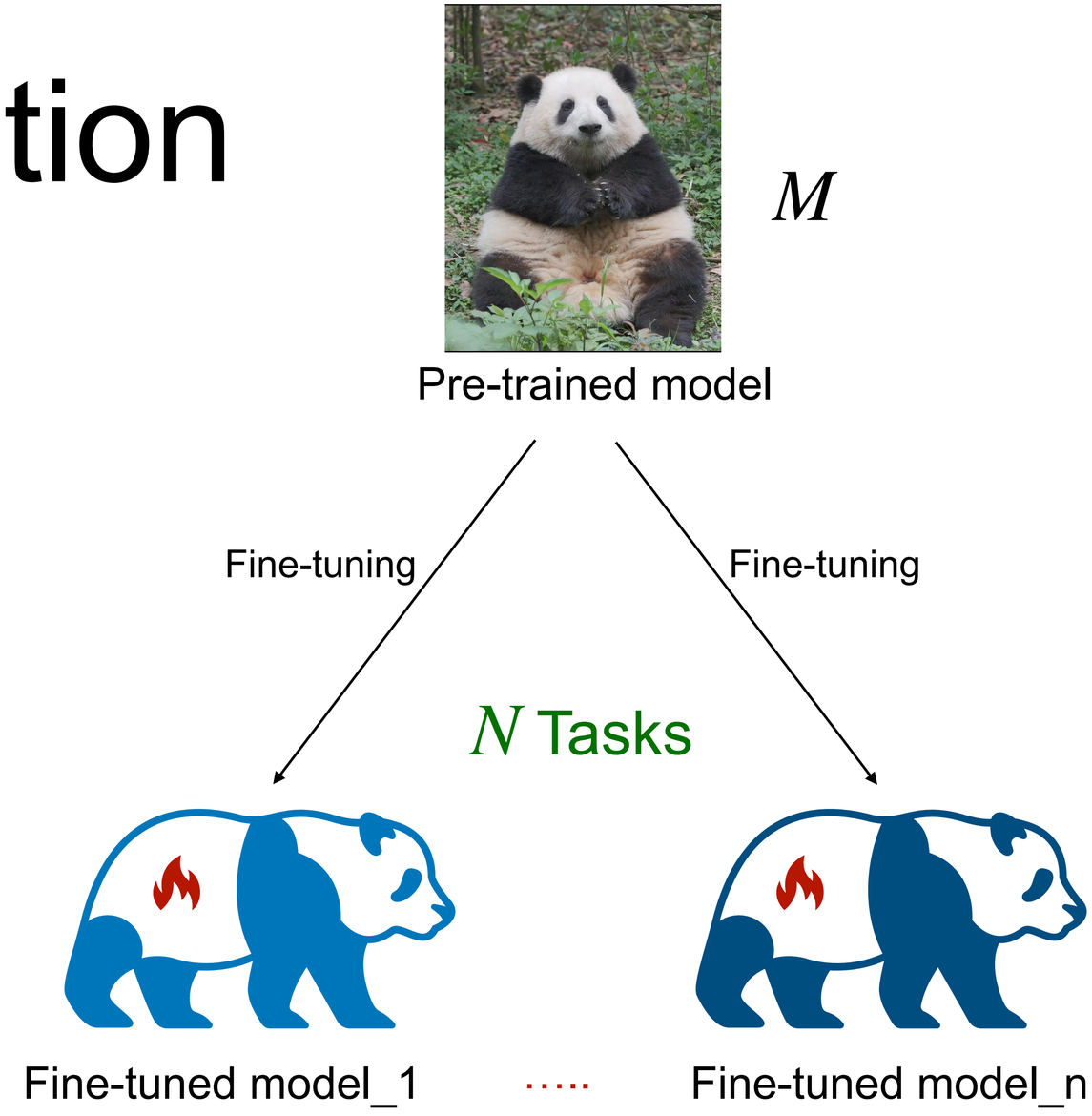
RST-LoRA improves long document summarization by integrating **rhetorical structure theory** into the LoRA model, outperforming previous methods.

Motivation

- Why do we need **low-rank** approximation?
- Why do we need **discourse** knowledge?

Motivation

- Why do we need low-rank approximation?
  - Model size ↑ 👉 software and hardware ↑

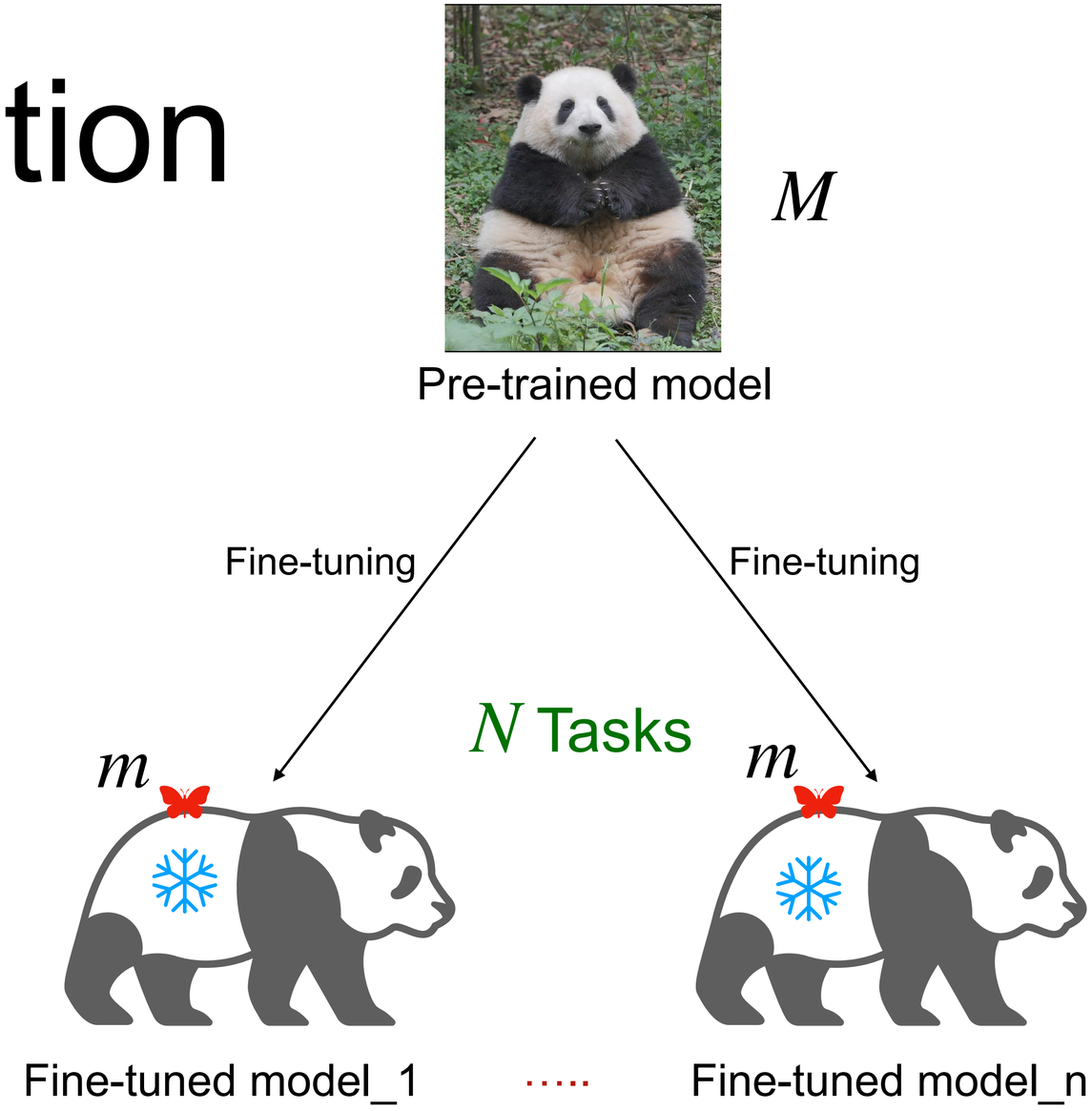


Vanilla full-parameter fine-tuning (FFT)

$N \times M$

Motivation

- Why do we need low-rank approximation?
  - Model size ↑ 👉 software and hardware ↑



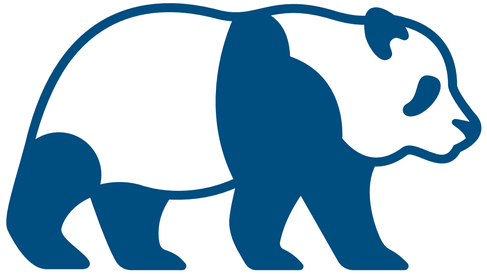
Parameter-efficient fine-tuning (PEFT)

$$N \times m + M$$

Motivation

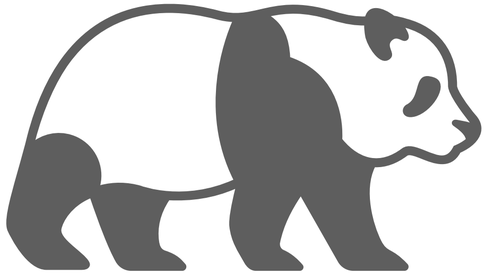
- Why do we need low-rank approximation?
  - Model size ↑ 👉🏻 software and hardware ↑
  - Only 0.01–1% of the parameters, PEFTs ≈ FFT

$N\times$



⩔

$N\times$ 🦋 $+$



FFT vs PEFT

Motivation

- Why do we need discourse knowledge?
  - Challenges in PEFTs
    - Latent text relations
    - Importance level of different sentences

Ghazvininejad et al. (2022); Zhao et al. (2023)

Reason

EPFTs are not driven or guided by discourse knowledge during the training phase, as this is not explicitly present in the input data.

RST Prerequisite

- Rhetorical Structure Theory (RST) is helpful for determining:
  - Which sentences **should or should not** be included in the summary
    - Sentences relations
    - Discourse importance level

RST Prerequisite

- **EDU1** is the most pivotal component
- **EDU2** provides information for **EDU3**
  - It is not a problem to delete EDU2
  - It is still fine to delete both EDU2 and 3

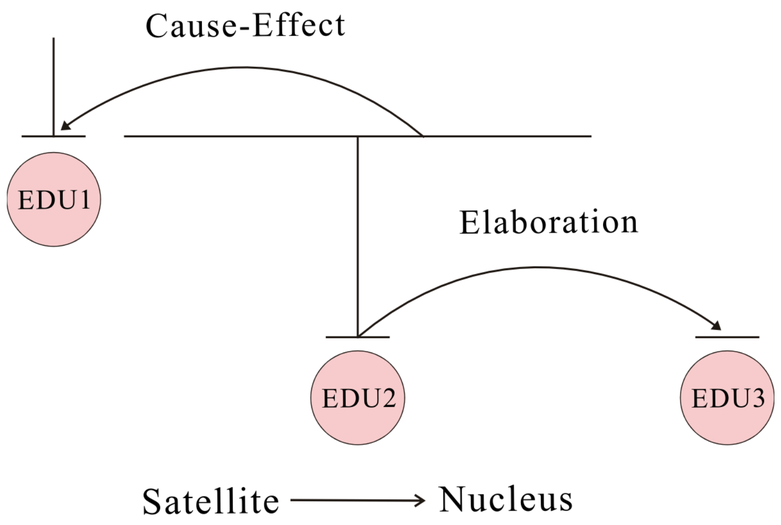


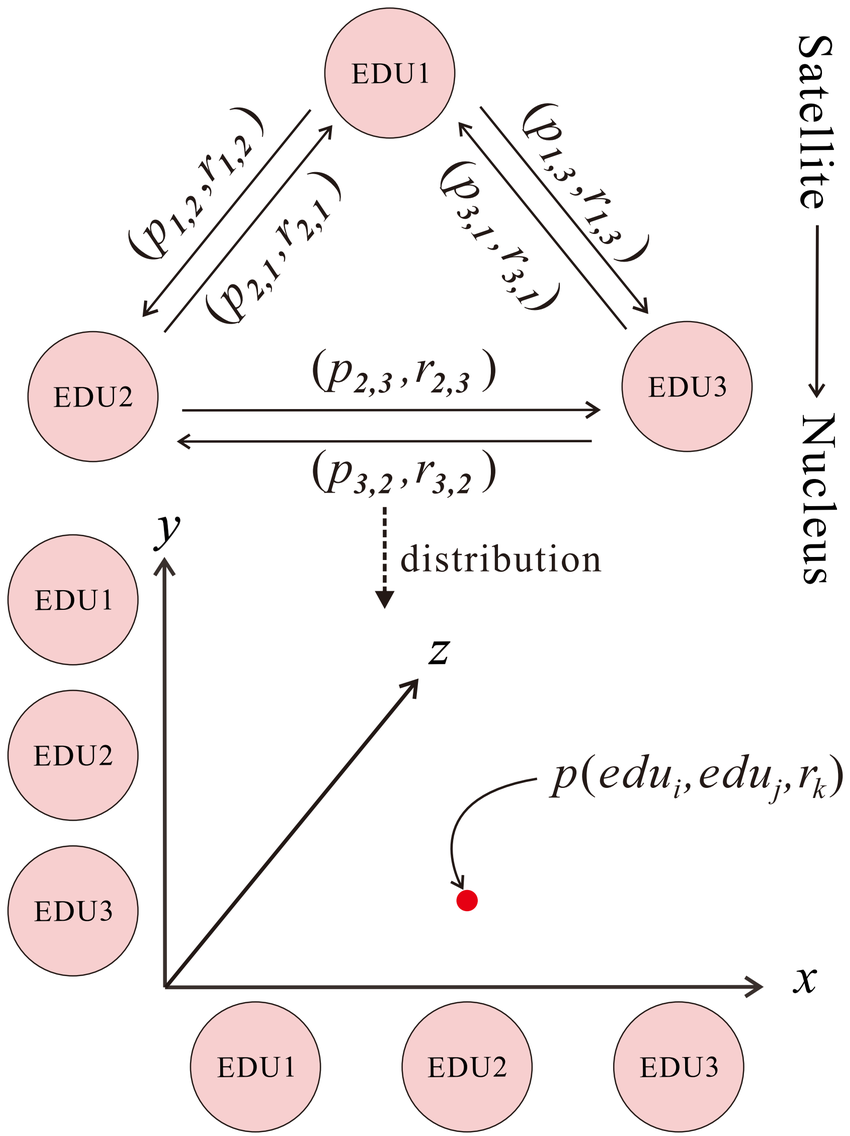
Figure 1: An example of RST tree: [*Utilizing discourse structure to enhance text summarization is beneficial.*]$^{\text{EDU1}}$ [*This technique can be used to identify key ideas and capture often overlooked nuances.*]$^{\text{EDU2}}$ [*Accurate capture of these complex structures facilitates the generation of good summaries.*]$^{\text{EDU3}}$

Our Method

- RST Distribution
- RST-Aware Injection

Our Method

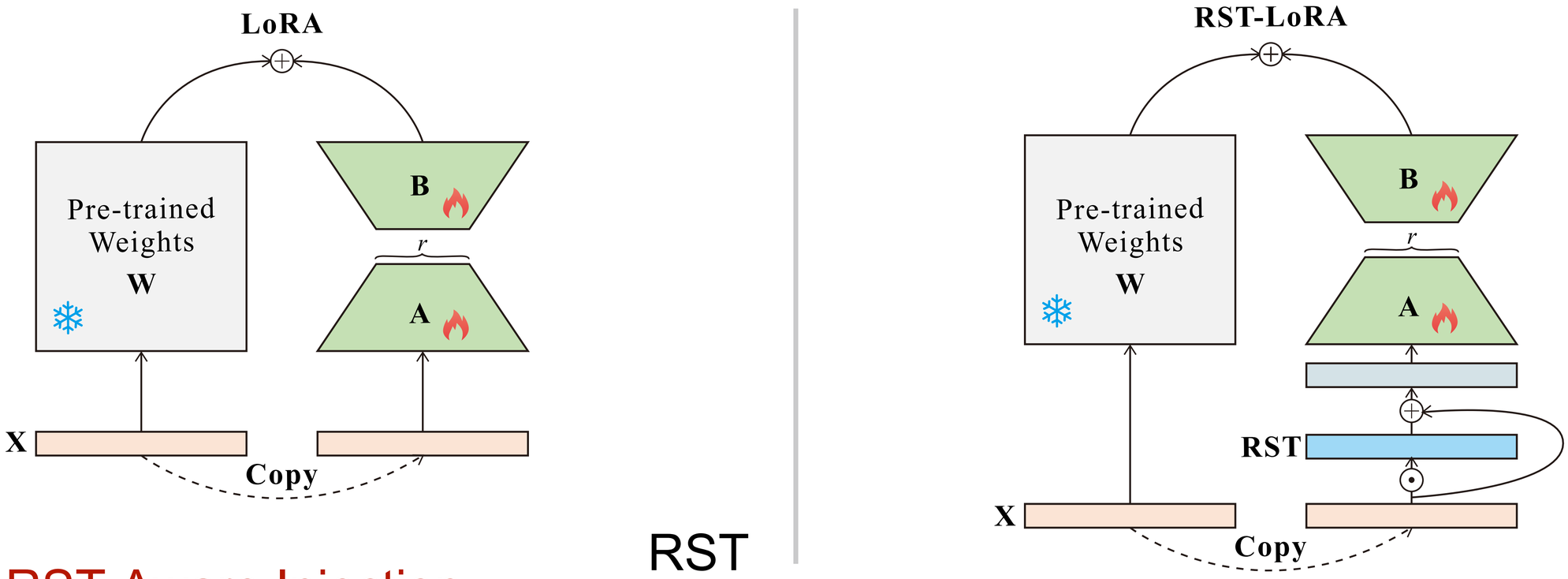
- RST Distribution
  - Each point indicates the probability value $p(edu_i, edu_j, r_k) \in [0,1] \subseteq \mathbb{R}$ that edu_i is the nucleus of edu_j with discourse relation r_k. (Pu et al., ACL 2023)
  - We average and merge the y-axis of the matrix, and the merged value $c(edu_i, \overline{edu_j}, r_k)$ is called the importance index of edu_i with relation r_k.



Our Method

- RST Distribution (4 variants)
  - RST^{b}_{wo}: Binary, label-agnostic representation (1 or 0)
  - RST^{b}_{w}: Binary distribution with relation labels
  - RST^{p}_{wo}: Label-omitted probabilistic representation
  - RST^{p}_{w}: Most fine-grained representation with relation types and probabilities

Our Method



RST

- RST-Aware Injection
  - $h \leftarrow h + X(W^{down}_{A\times r} W^{up}_{r\times B})$ (vanilla LoRA)
  - $h \leftarrow h + [(X \odot (1+\gamma))(W^{down}_{A\times r} W^{up}_{r\times B})$ (ours)

Experiments

- Experimental Settings
  - Datasets
  - Parser
  - Metrics
  - Training and Inference

Experiments

- **Datasets**
  - Multi-LexSum (ML, Shen et al., 2022)
  - eLife (Goldsack et al., 2022)
  - BookSum Chapter (BC, Kryscinski et al., 2022)

Experiments

- **Parser**
  - DMRST (Liu et al., 2020, 2021).
  - Extracting probabilities and type labels from final logits layer

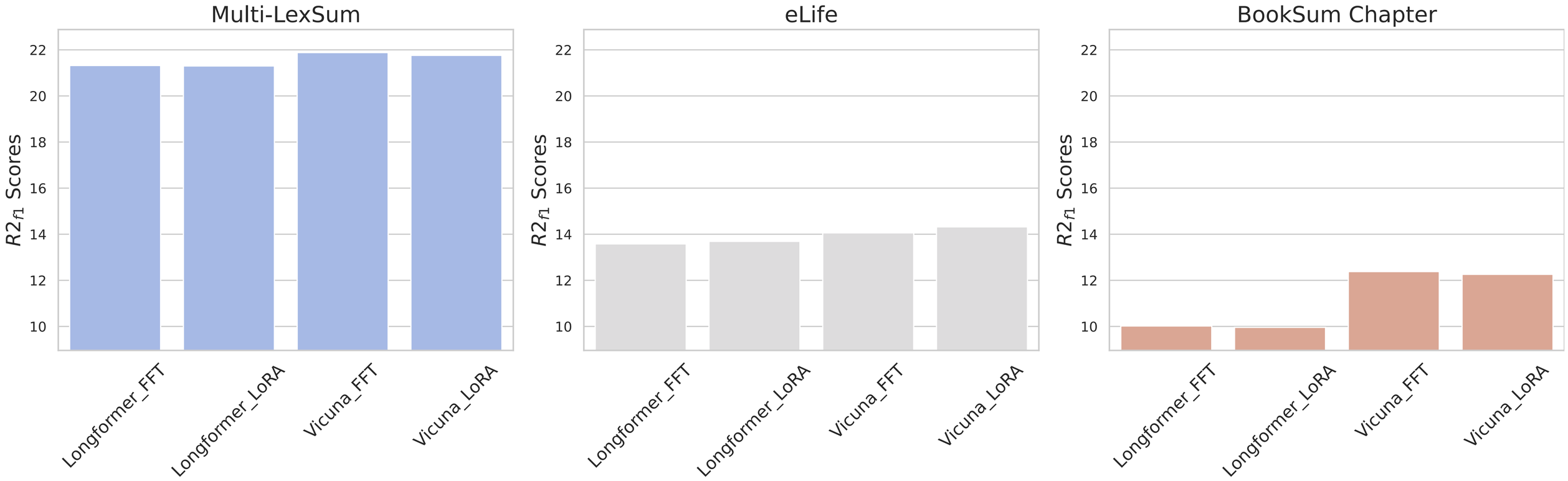
Experiments

- **Metrics**
  - F1 scores of Rouge-1 (R1), Rouge-2 (R2), Rouge-L (RL), and Rouge-Lsum (RLsum) (Lin, 2004)
  - BERTScore (Zhang et al., 2020)
  - METEOR (Banerjee and Lavie, 2005)
  - sacreBLEU (Post, 2018)
  - NIST (Lin and Hovy, 2003)

Experiments

- **Training and Inference**
  - Backbones
    - Longformer (Beltagy et al., 2020) 👉🏻 Seq2Seq
    - Vicuna13B-16k (Zheng et al., 2023) 👉🏻 GPT
  - Baselines
    - Backbones w/ FFT
    - Backbones w/ LoRA
    - GPT-4 (in-context learning)
  - Other SOTAs

Main Results



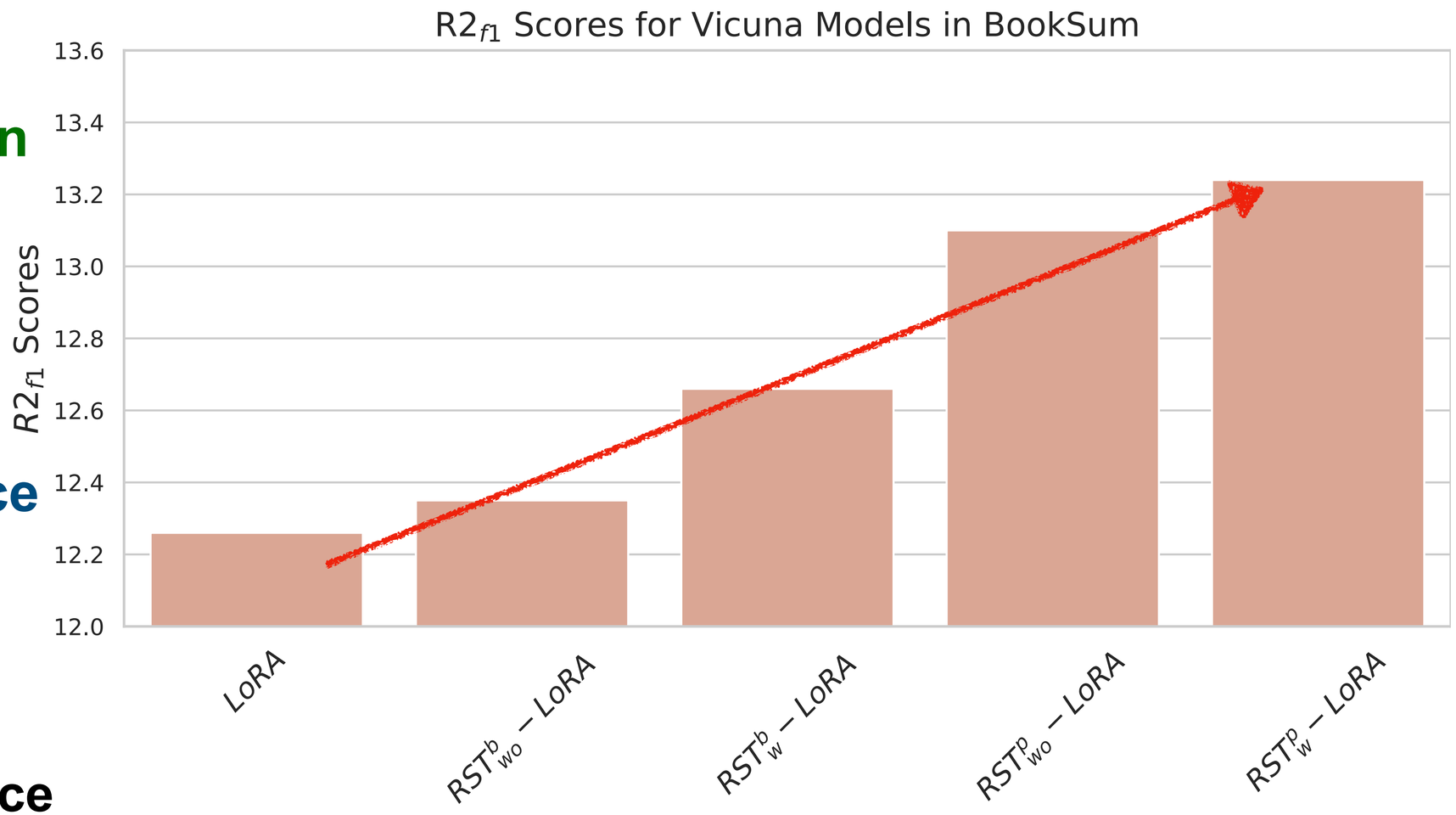
LoRA vs. FFT: Comparable, more efficient

RST variant performance

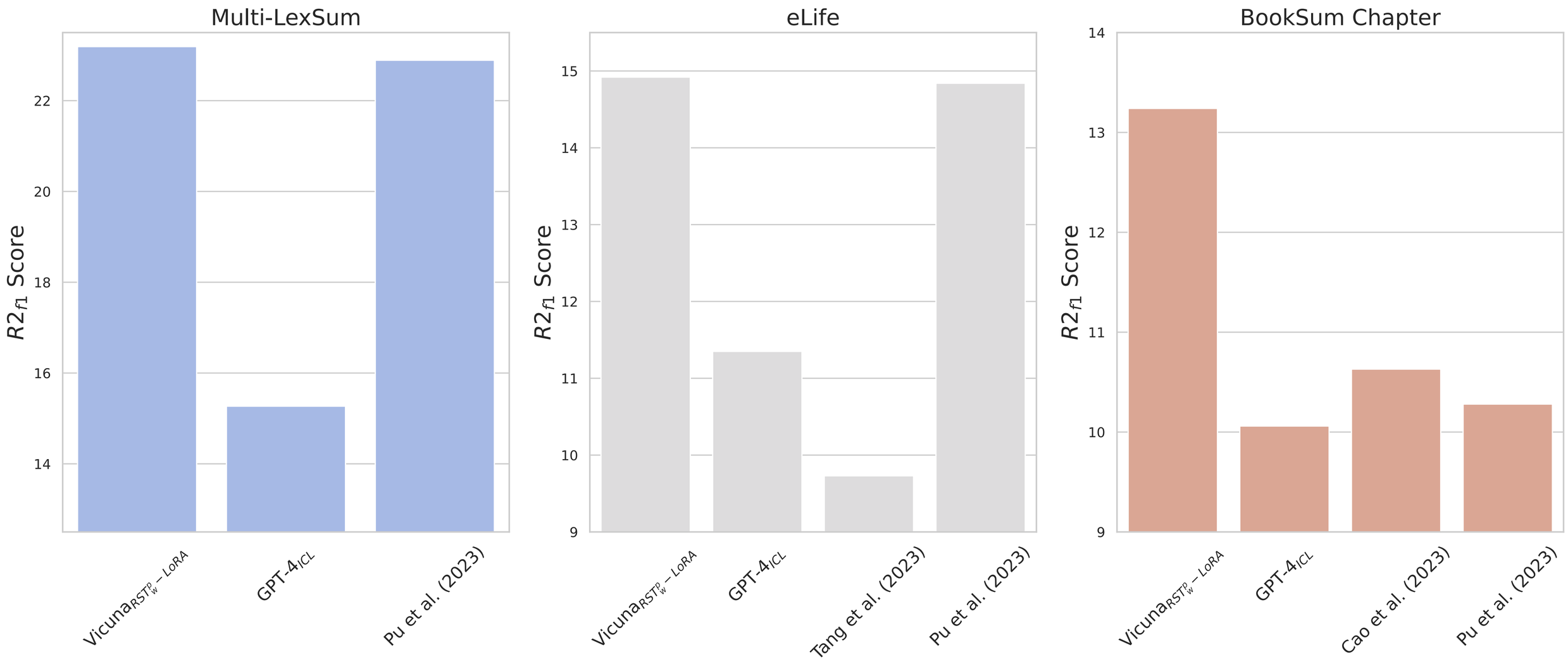
- **Label integration**
- **Uncertainty consideration**

Both complementarily enhance model performance

RST^{p}_{w}**-LoRA: Best performance**



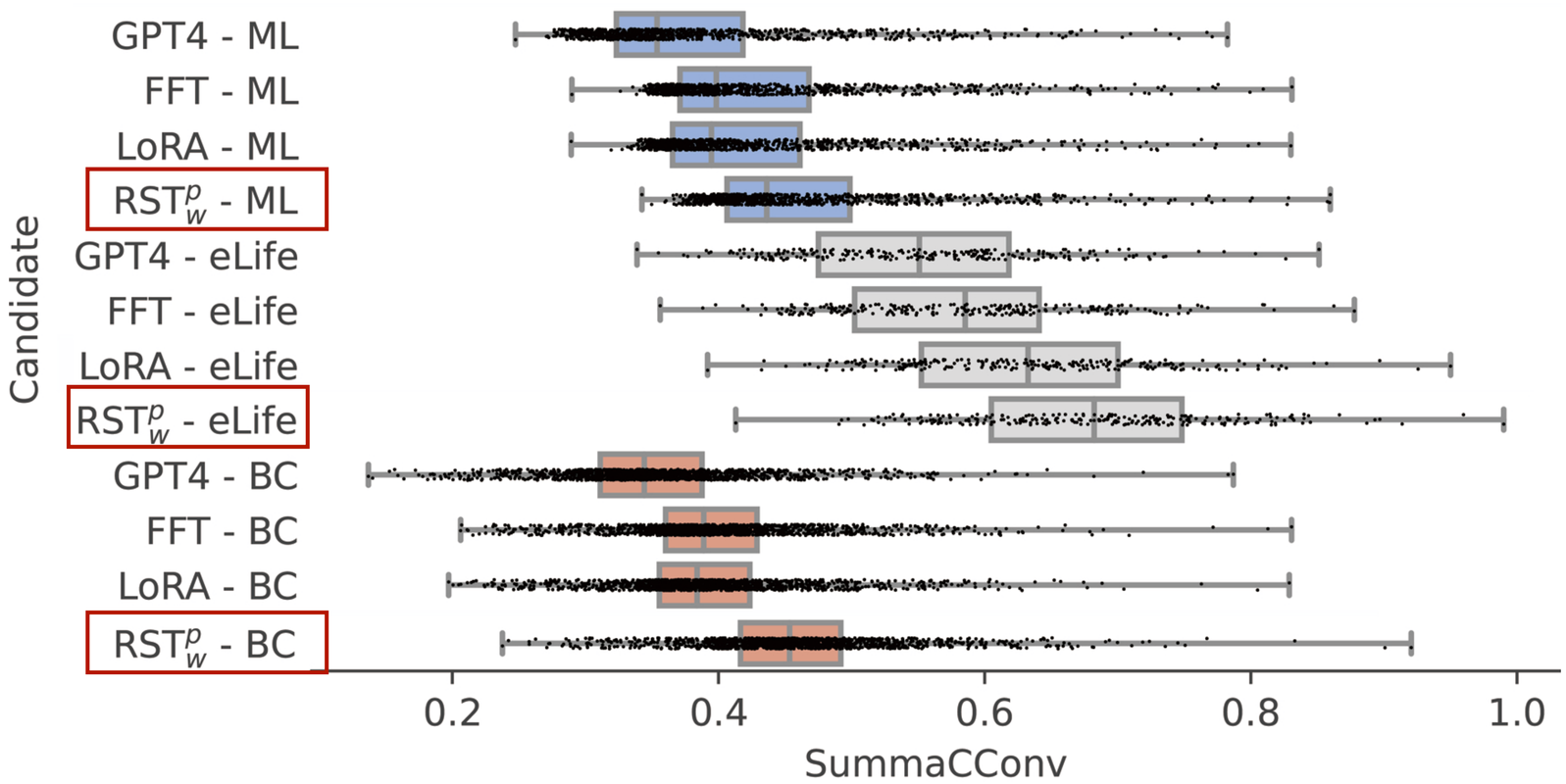
Our best model vs GPT-4 and SOTAs



Hallucination Checking

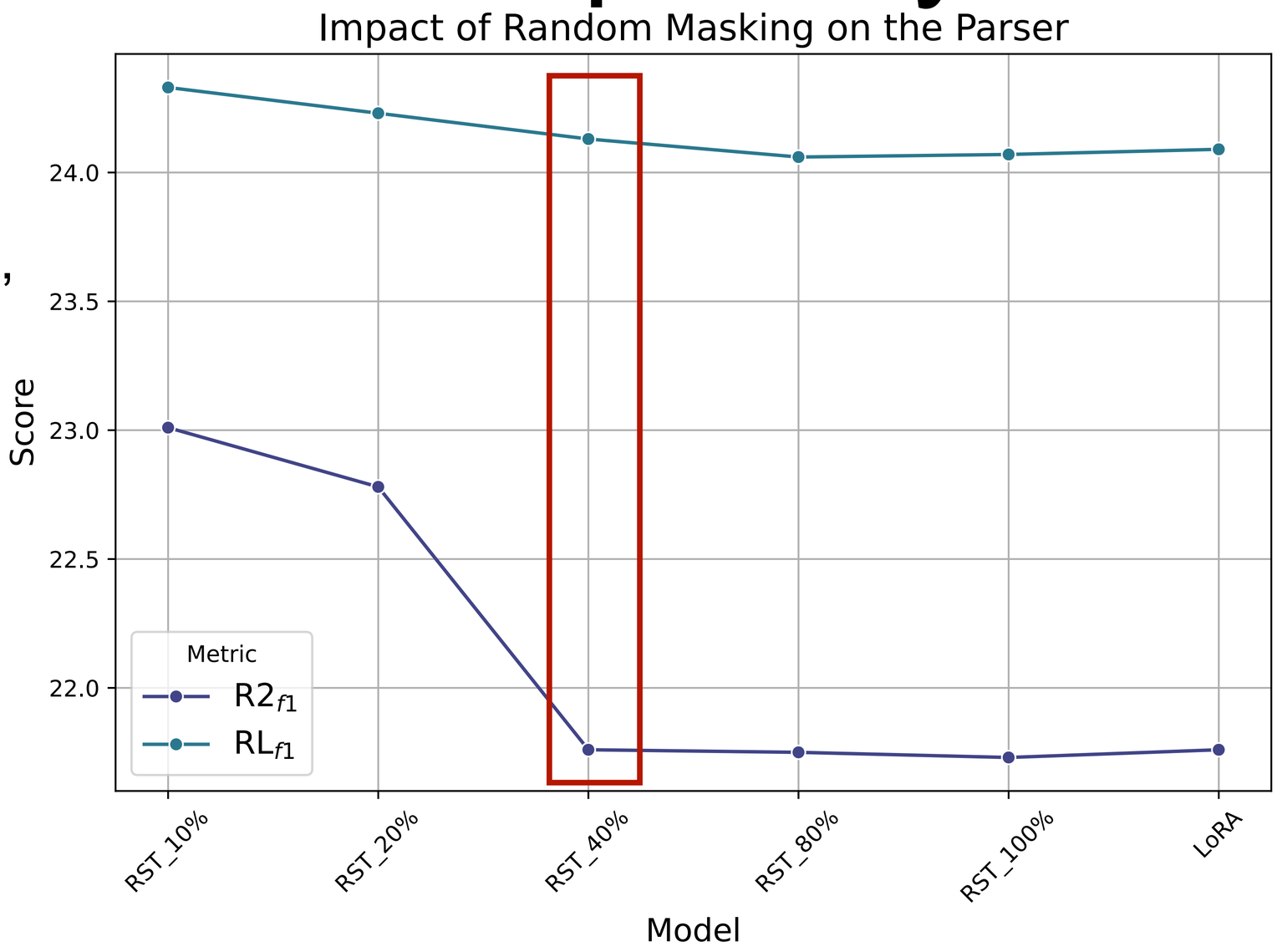
SummaC testing: 0-1 score range

- **GPT-4:** Weakest consistency
- **RST enhances LoRA:** Reduces hallucinations

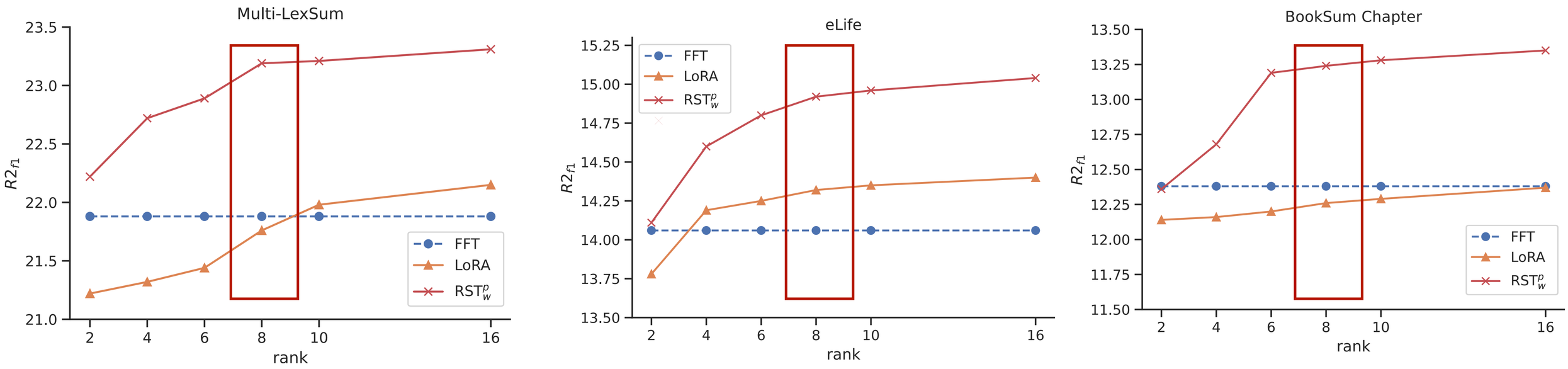


Impact of Parser Capability

- **Parser impact test:** 10%, 20%, 40%, 80% masking
- **Vicuna backbone:** Multi-LexSum dataset
- **Performance declines:** >40% noise



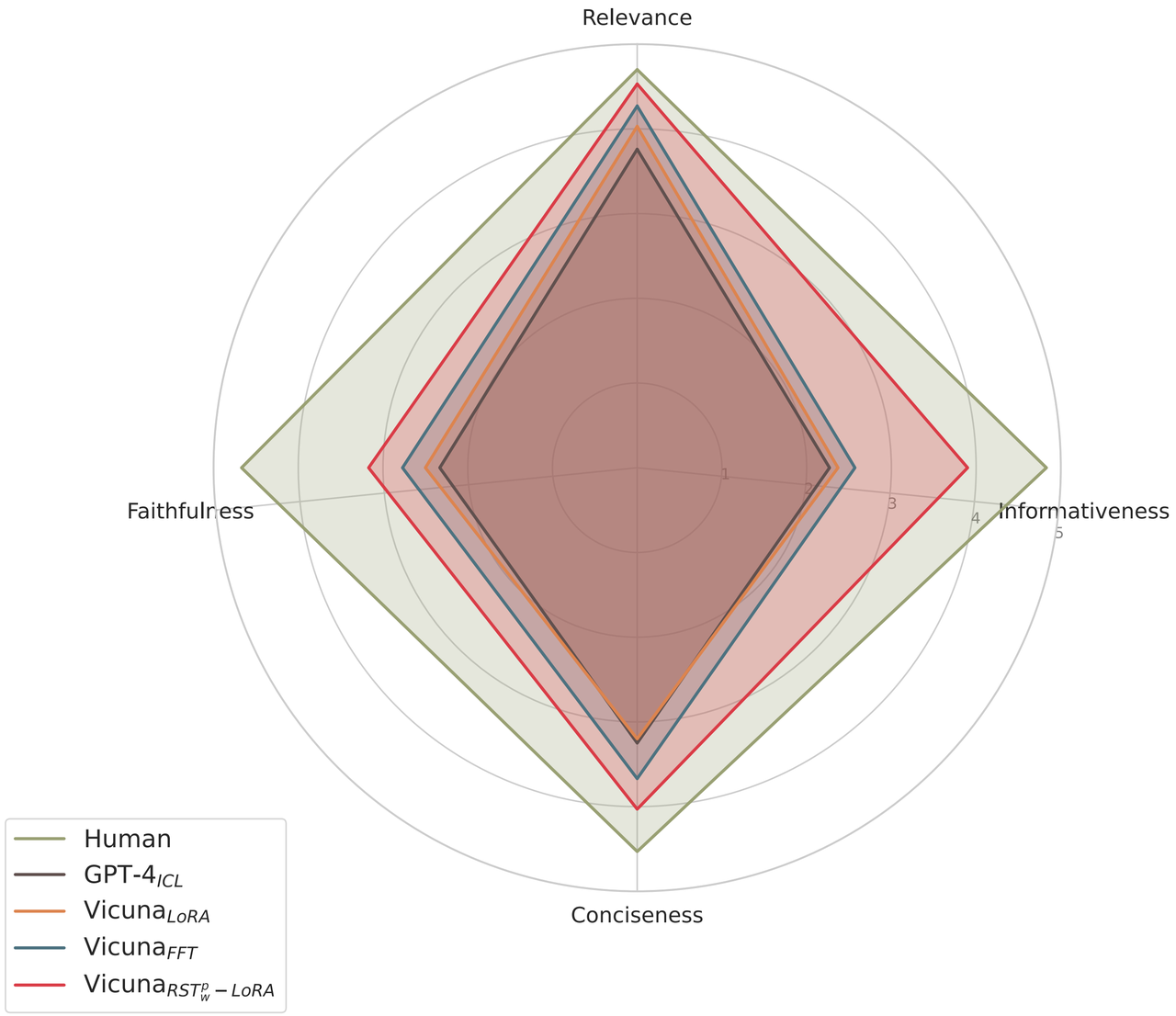
Impact of Different Rank r



$r = 8$ is a trade-off point between performance gain and computational cost

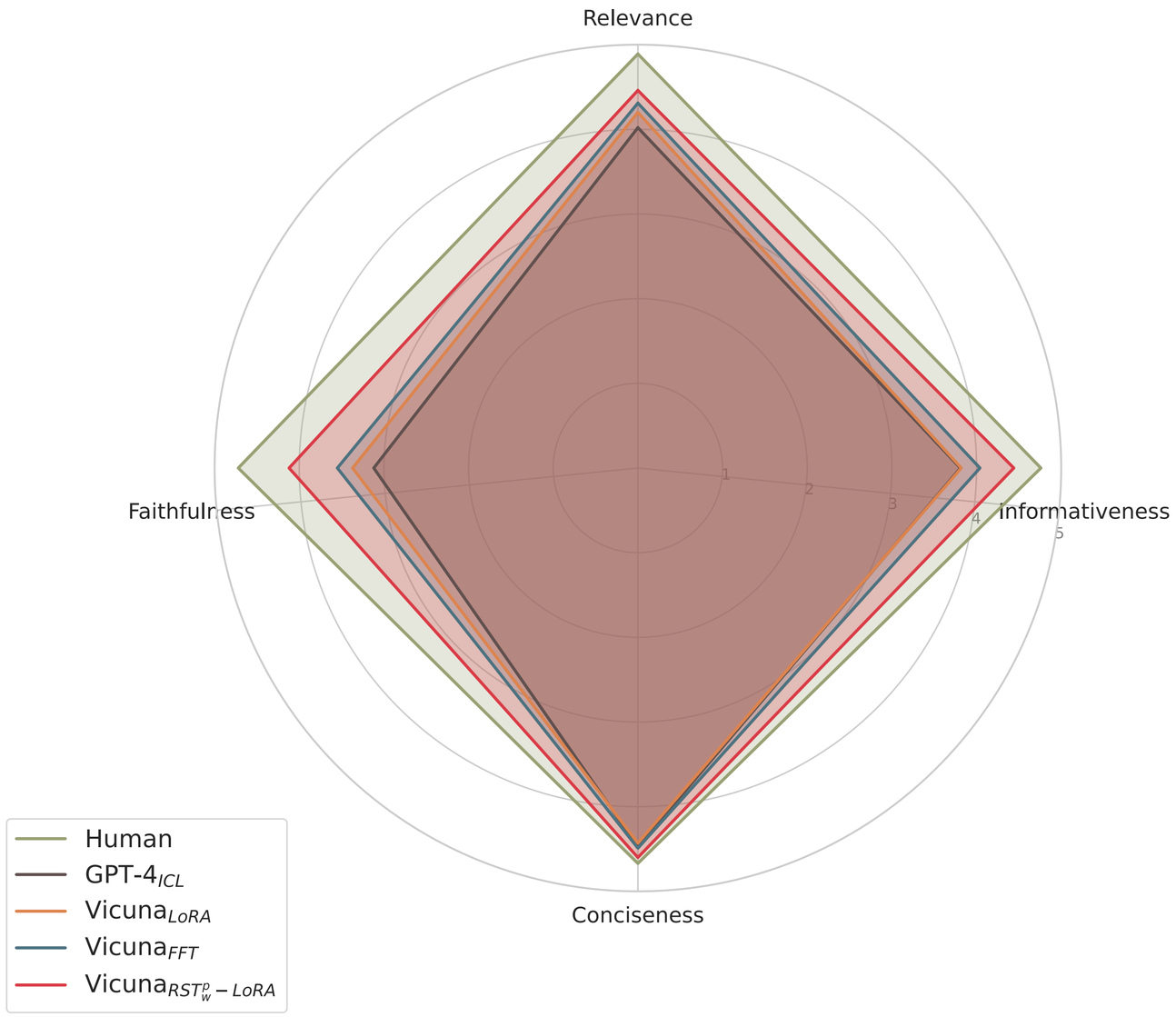
Human Evaluation

- **Human evaluation:** BookSum, 10 instances
- **Evaluators:** CL/CS Graduate candidates, blind test
- RST_w^p**-LoRA:** Highest neural model performance



GPT-4 Evaluation

- **GPT-4 self-evaluation:** Lowest scores to own answers
- RST_w^p**-LoRA:** more closer to the quality of human-generated summaries



Conclusion

- A method for injecting **discourse knowledge** into the training of LoRA model.
- Discourse uncertainty and relation labels are **complementarily**.
- Our model **outperforms** current SOTA models in specific evaluation metrics.

More Info

- **Code:** https://dongqi.me/projects/RST-LoRA
- **Questions:** dongqi.me@gmail.com

Thanks for listening

Q&A

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (Grant Agreement No. 948878)