

You're reading a paper that refers to an "LM-adapted T5." What does this mean?

has an embedding layer that is trainable

A model is built by adding an adaptor to T5 in order to make it better at some specific task.

T5 base model modified using adapters for Language modeling

language model loaded from T5 checkpoint?

some method to fine-tune the t5 model for a specific language task

it's customized for a specific task.

Some kind of Language Model (LM) and (T)ransformer

It means it uses a unified transformer architecture (encoder & decoder)

You're reading a paper that refers to an "LM-adapted T5." What does this mean?

fine tuned T5 model for
generating texts

T5 was pre-trained with a mask-infilling objective. It never saw targets that were full complete sentences. But, we can "adapt" it to language generation by doing further training to predict a sequence suffix given a prefix.

What are some reasons why a RAG system might perform worse at responding to a query than an LM system without retrieval?

incorrect data in db

retrieval errors, irrelevant documents, or misalignment

the RAG was trained on a different domain

the retrieved documents are noisy or contain more information than needed

Retrieved documents have misleading information

Rag may provide context that is not relevant that confuses the model

out of domain query for the particular retrieval domain

When the context is distracting.

What are some reasons why a RAG system might perform worse at responding to a query than an LM system without retrieval?

The retrieved data can be irrelevant or distracting.

If the retriever sucks or gives useless retrieving

The retrieved information can contain incorrect/irrelevant info, which introduces noise

outdated data in database

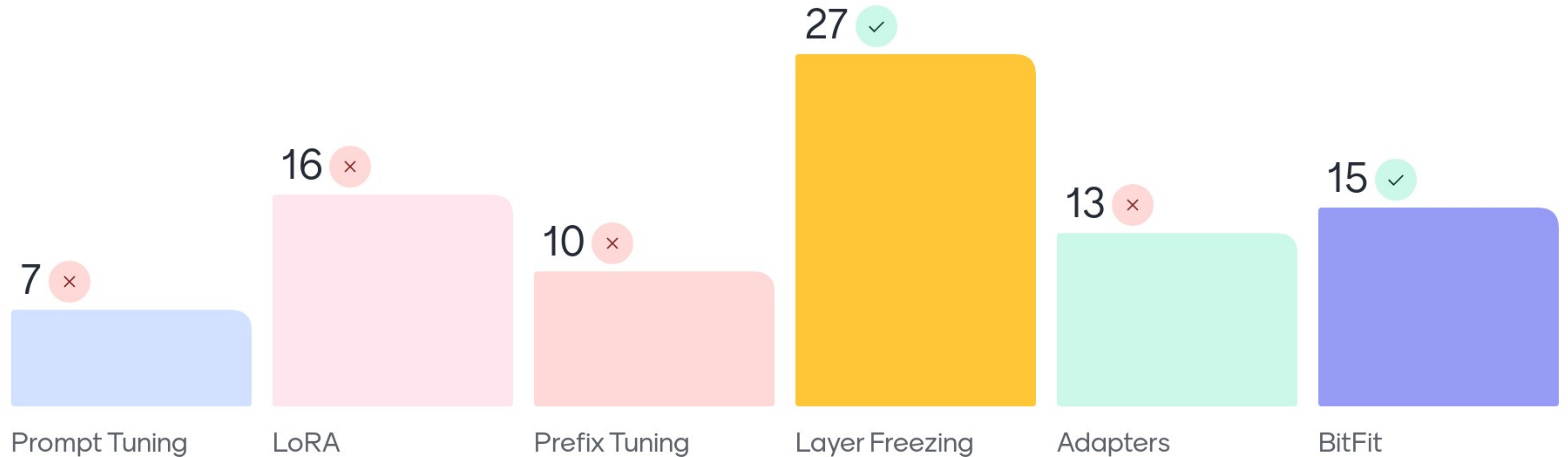
The answer of some responses could be wrongly

retrieving irrelevant data

Getting the incorrect/misleading/outdated data from database.

When retrieval isn't really required

What of the following param-efficient finetuning methods involve picking a small subset of the weights and just tuning those?



How many examples should you use for few-shot learning?

it depends

1-5

Few

a few (3-5)

few - 1,2?

The minimum number that
allows good performance

5

1 to 3

How many examples should you use for few-shot learning?

Whatever amount gets the best results :)

≥ 2

2-5

any amount that can fit into context length

1-3

1 or 2, no more than 3

more than one

5

How many examples should you use for few-shot learning?

More than 1, but not too many

2-5

anything more than 1

3 - 5 examples

Up to 5

3-4

Few

5-6

How many examples should you use for few-shot learning?

depends on your budget, the context window size, the size of examples, ...

it depends