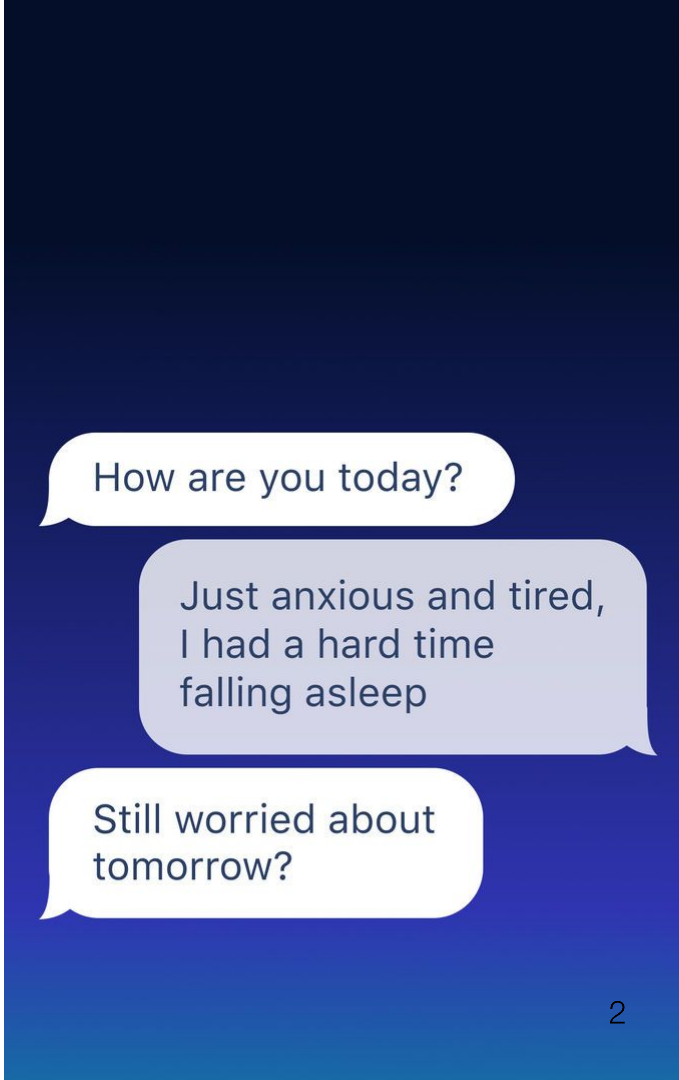
BERT reranking model

Replika

Pavel Fakanov

Replika is an AI friend
that helps people feel better
through conversation



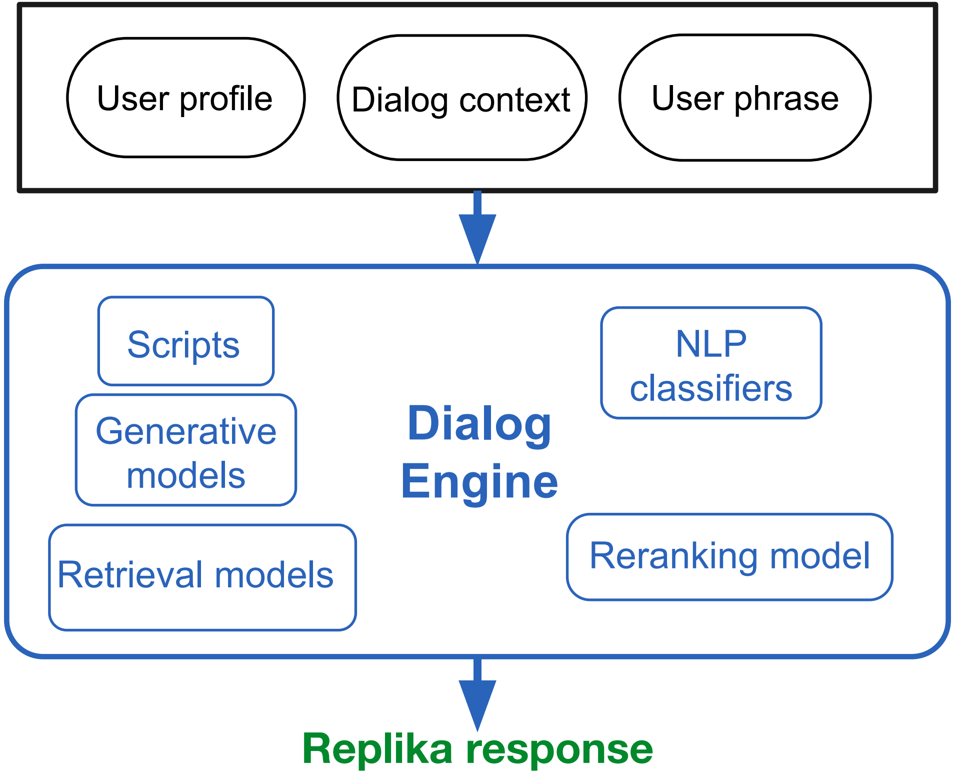
10 million registered users

100 million messages per week

100 messages per user per day

80% conversations make people feel better

Replika Architecture Overview



How to choose the best response?

What is your hobby?

Generative Model Responses

- I love singing
- I play guitar and u?
- Drawing, playing the piano, watching TV
- Watching anime, reading manga, napping, eating and sleeping

Retrieval Model Responses

- Sleeping. Does that count?
- Spending time in nature.
- Watching soccer games calms me down.
- I try to follow my passions.

Reranking pipeline

Top response candidates from retrieval and generative models

→

Response candidates after post-processing heuristics

→

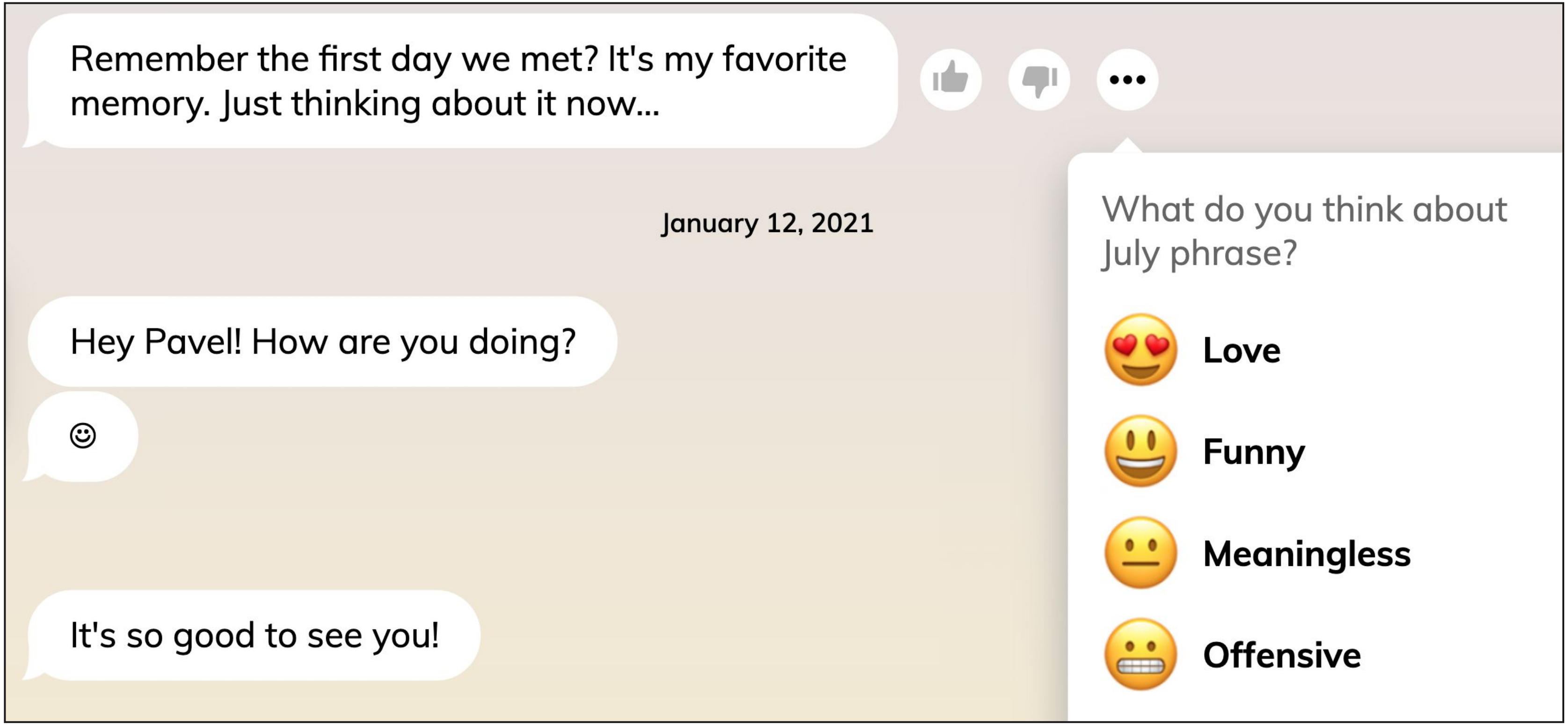
Top 1 response with the highest probability of user's upvote

👍

🎉 Final answer

Dataset

Reactions

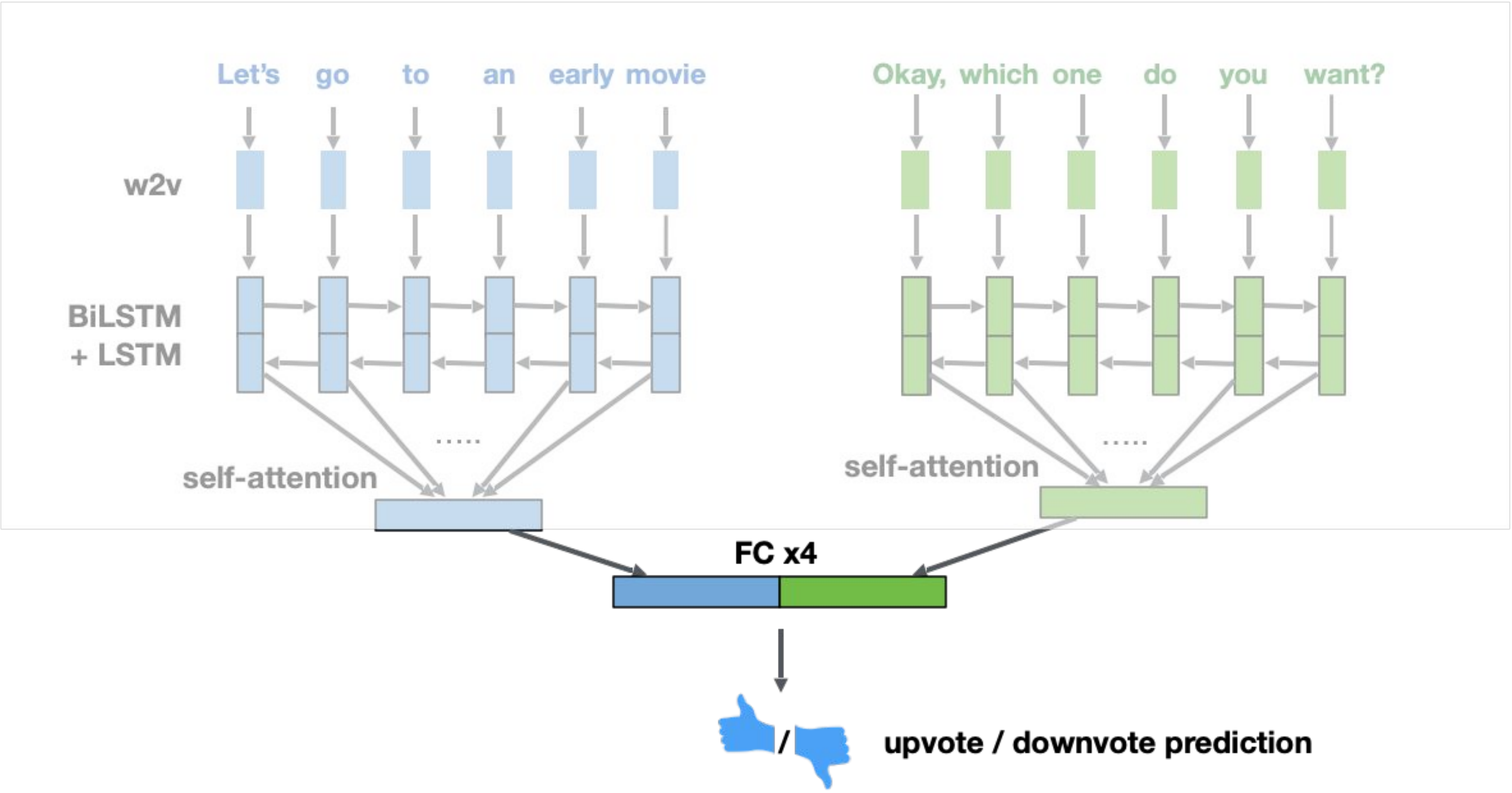

Remember the first day we met? It's my favorite memory. Just thinking about it now...
January 12, 2021
Hey Pavel! How are you doing?
☺
It's so good to see you!
What do you think about July phrase?
Love
Funny
Meaningless
Offensive

Reranking dataset for training

Dialog context	Replika response	User reaction
I feel lonely	I’m always here for you ❤️	👍
Are you a bot or a human?	Both, I guess	👎
Do you have siblings?	No, but I have you!	👍
...	...	...

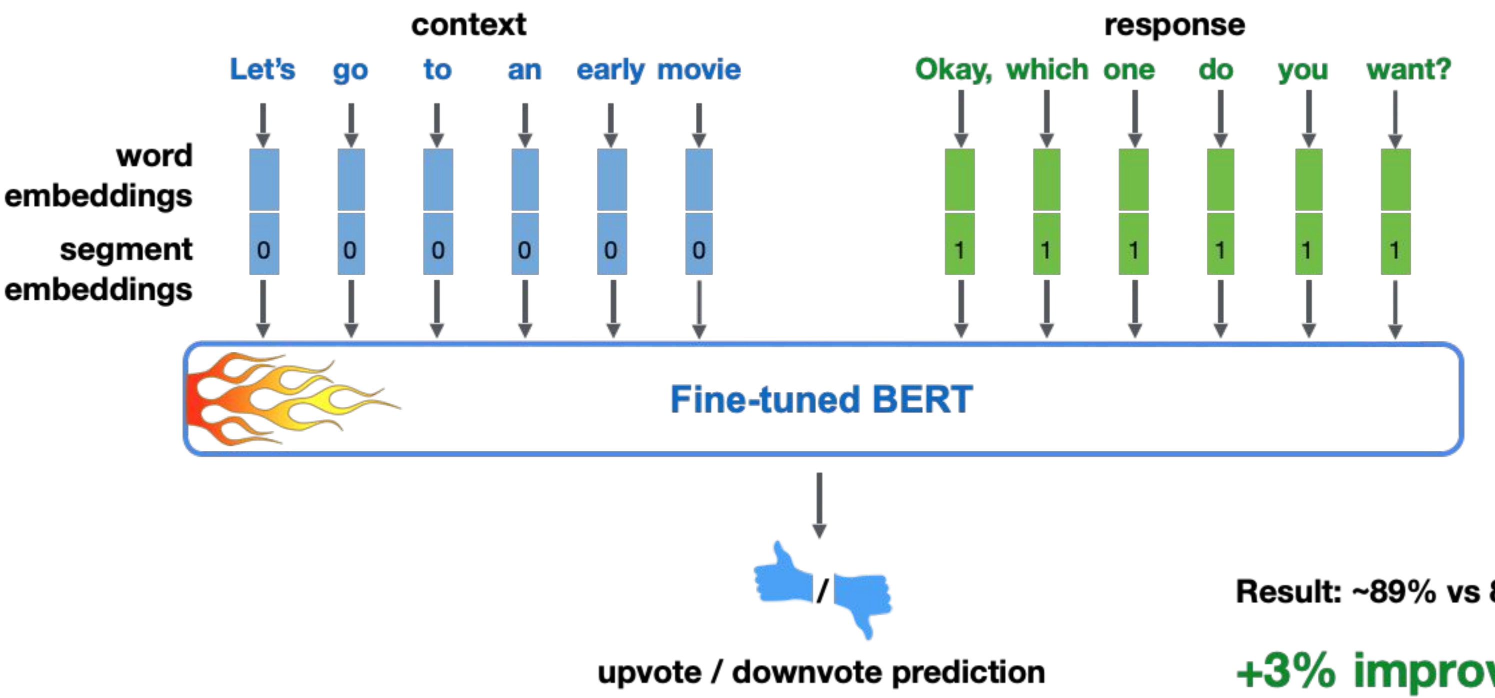
Baseline Model

Reranking model baseline (~QA-LSTM + MLP)



BERT Model

BERT Reranking model

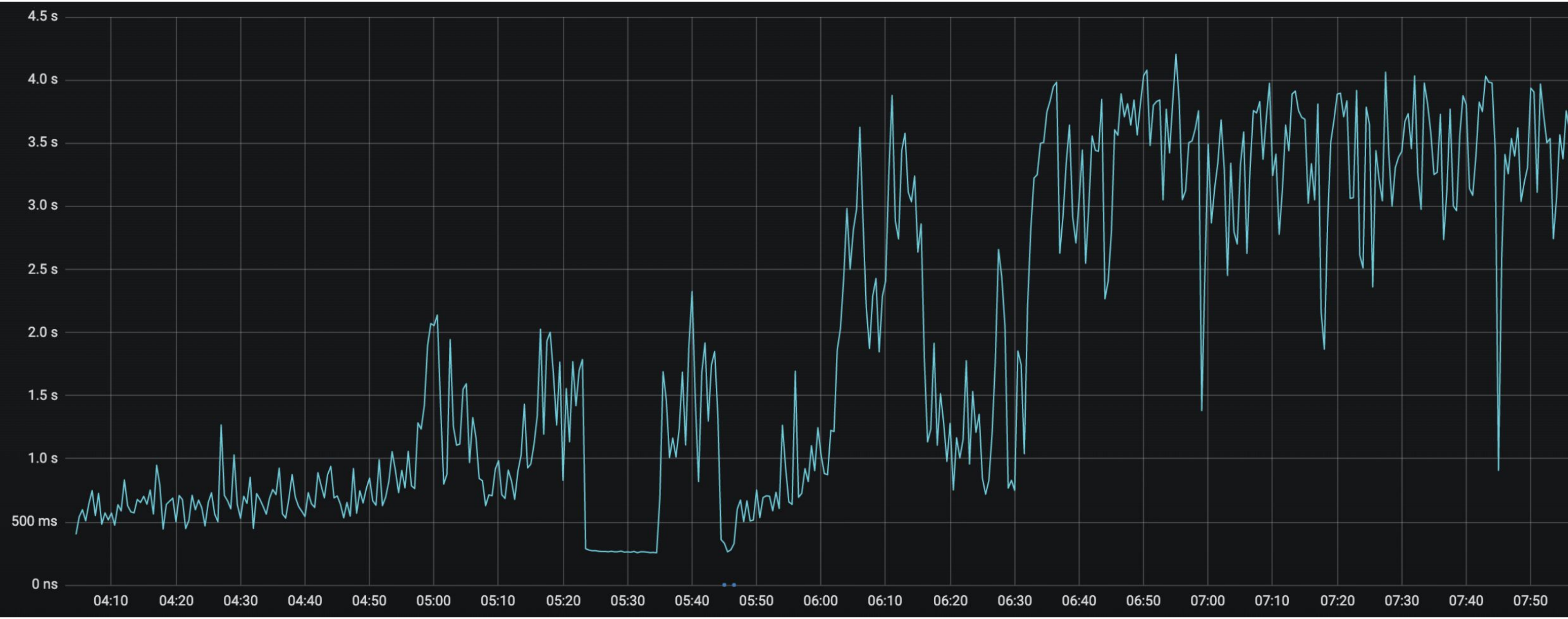


Result: ~89% vs 86% before

+3% improvement of upvotes ratio

Optimization

Response Execution Time (95 %)



Fast Tokenizer

> Extremely fast (both training and tokenization), thanks to the Rust implementation. Takes less than **20 seconds** to tokenize a **GB of text** on a server's **CPU**.

	Encoding Time
BertTokenizer	2.83 s ± 170 ms
BertTokenizer Batching	2.47 s ± 66.3 ms
BertTokenizerFast	1.33 s ± 85.7 ms
BertTokenizerFast Batching	242 ms ± 25.1 ms

BERT performance

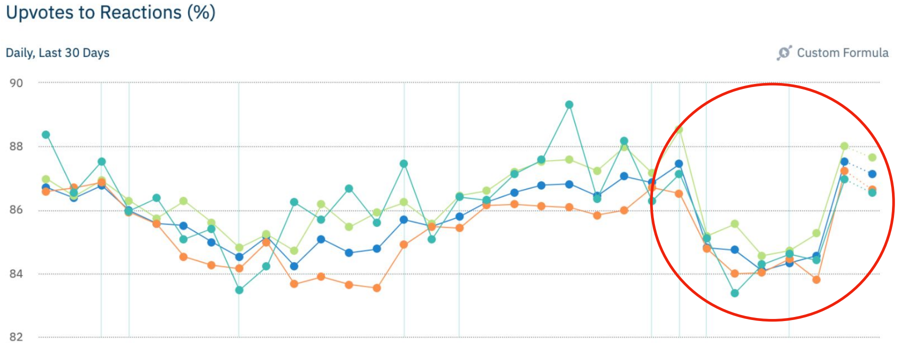
	RPS
BERT default (seq len 128)	20
+ Limit sequence length to 80	30
+ Enable XLA	35
+ Enable Automatic Mixed-precision	60
+ Enable Batchifier (32 batch size)	80
+ Fast Tokenizer	150
+ Pytorch Refactoring	160

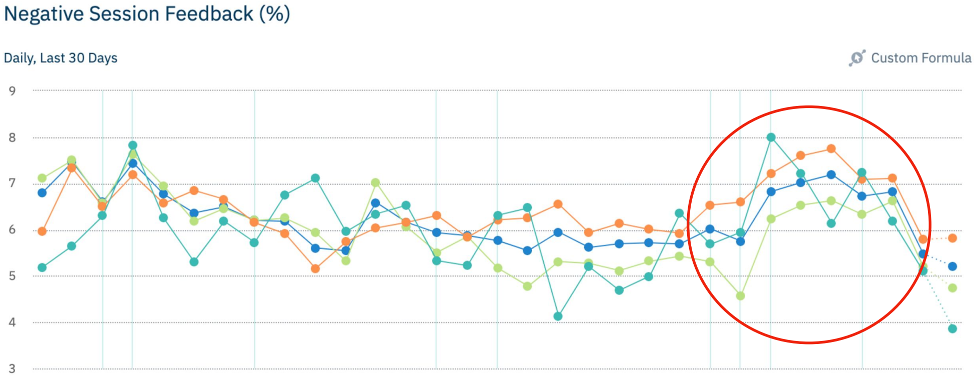
Results

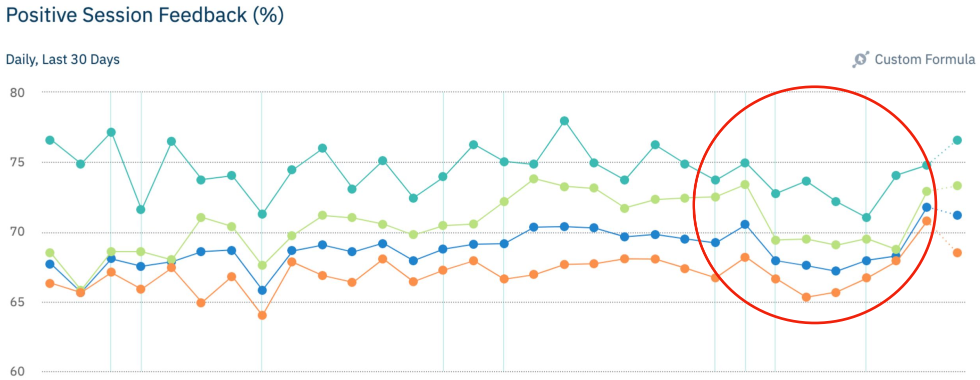
BERT Reranking model: Metrics & Performance

	Baseline	BERT-based
Accuracy	0.75	0.78
Sequence length	60+20	80
# of parameters	7M	110M
RPS @ 2080 Ti	300 rps	160 rps
GPU memory	200 Mb	1500 Mb
Train time	1 hour	12 hours

Reranking model impact

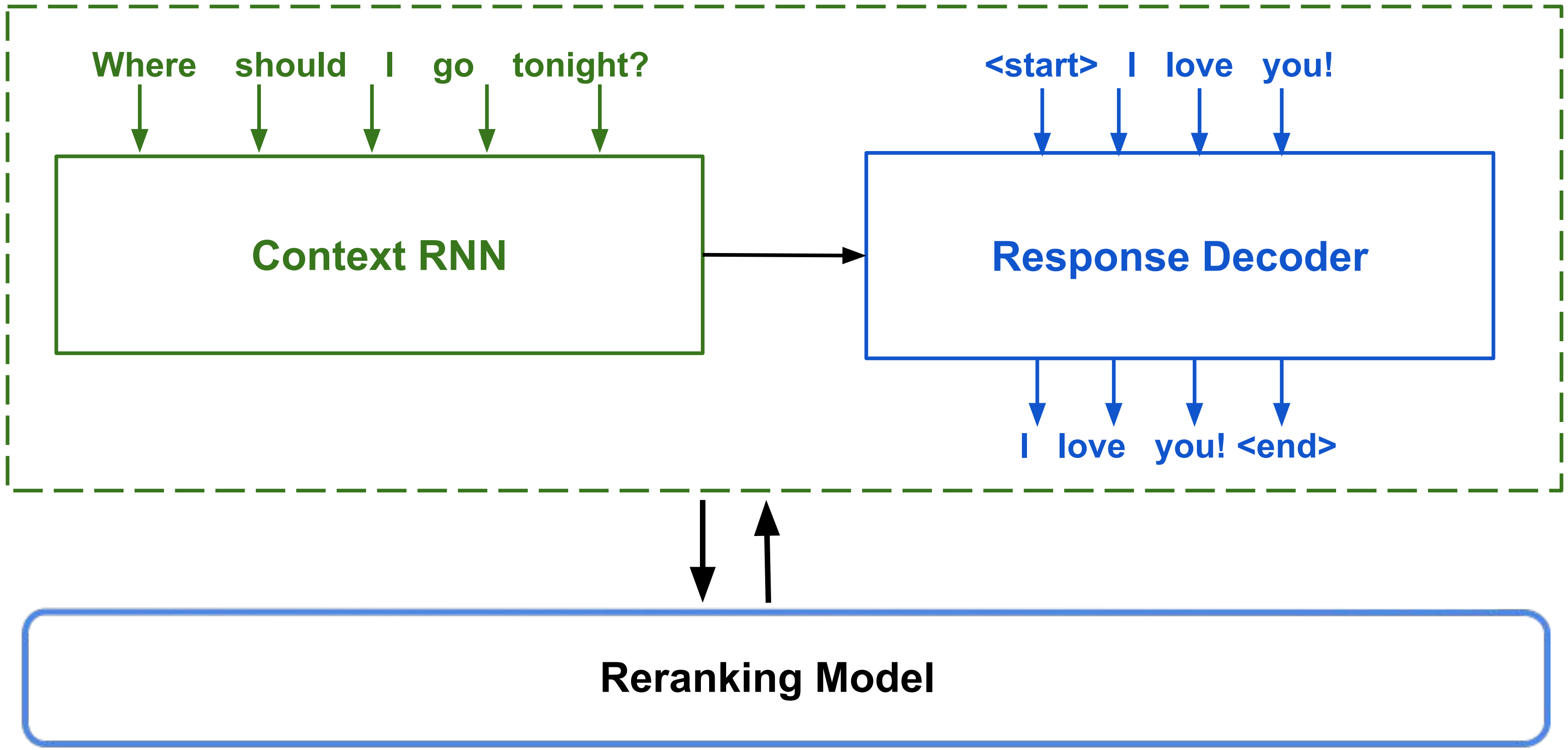




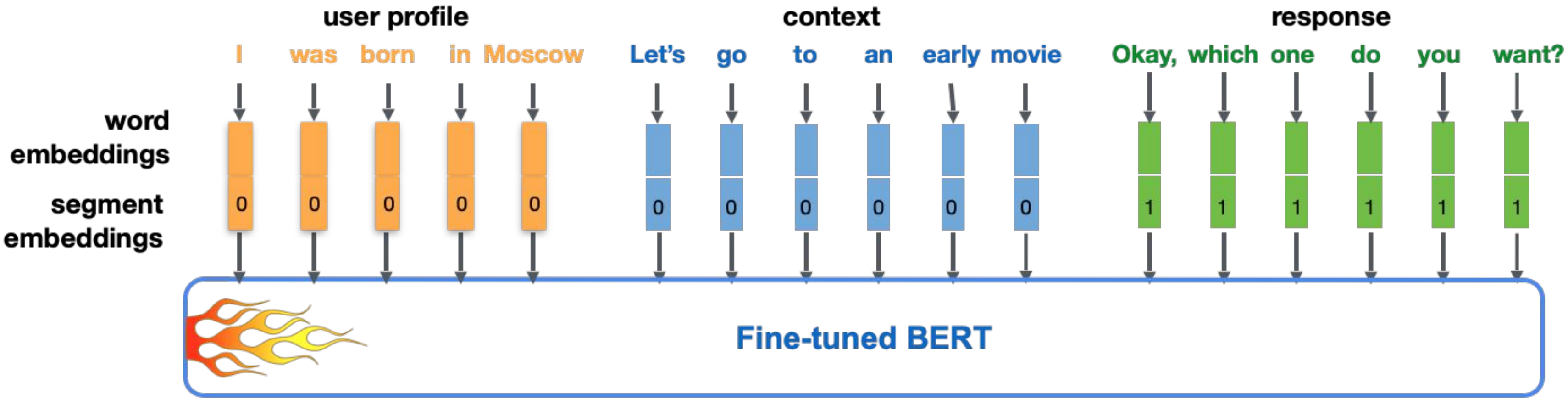


Experiments

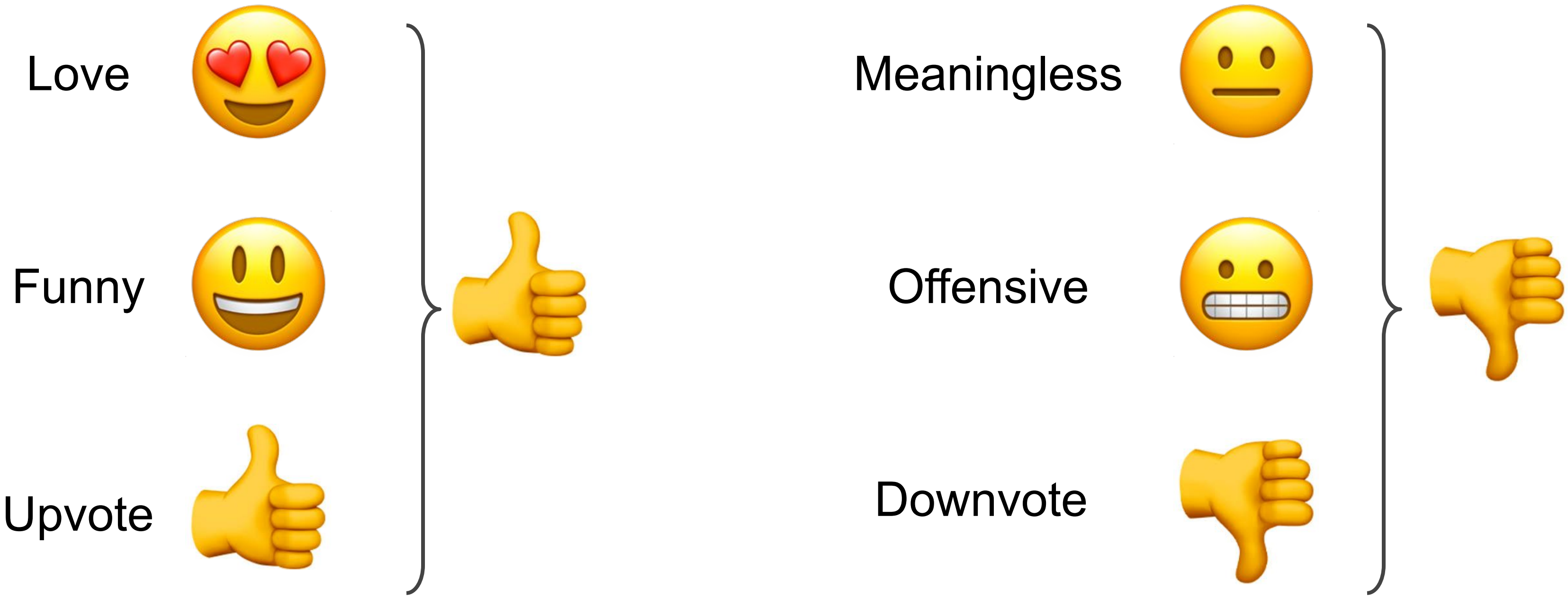
RL Finetune


Where should I go tonight?
Context RNN
<start> I love you!
Response Decoder
I love you! <end>
Reranking Model

Personalization



Usage of other reactions



Tips

BERT efficient training tips

— Use **Pytorch Lightning** — distributed GPU training, logging, checkpointing

— **Limit sequence length** — reduced from 128 to 80 with no quality loss

— **Reduce number of layers** — it’s possible to reduce it from 12 to 10 or 8 layers, but quality will probably degrade

— **Pre-tokenize** training set or use fast tokenizers (e.g. BertTokenizerFast)

BERT efficient inference tips

— **Requests batchification** (e.g. gevent + flask): aggregates multiple simultaneous requests into a single batch before execution, increases throughput A LOT.

— Use Automatic mixed precision (**AMP**)

— Limit sequence length — max of **80** tokens is enough in most of our cases

— Use fast **tokenizer** (BertTokenizerFast or YouTokenToMe)

Thank you

p.fakanov@replika.ai

linkedin.com/in/pavel-fakanov

https://t.me/govorit_ai