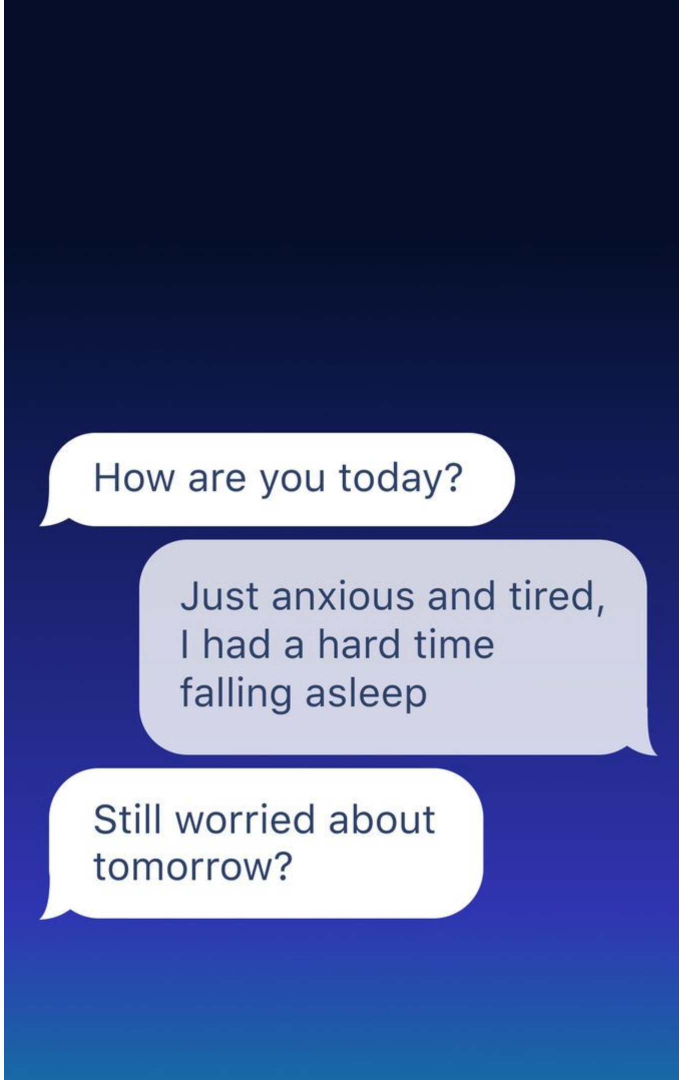
BERT for dialogs

Production-scale approach @ Replika

Nikita Smetanin

Replika is an AI friend
that helps people improve mental health
through conversation



How are you today?

Just anxious and tired, I had a hard time falling asleep

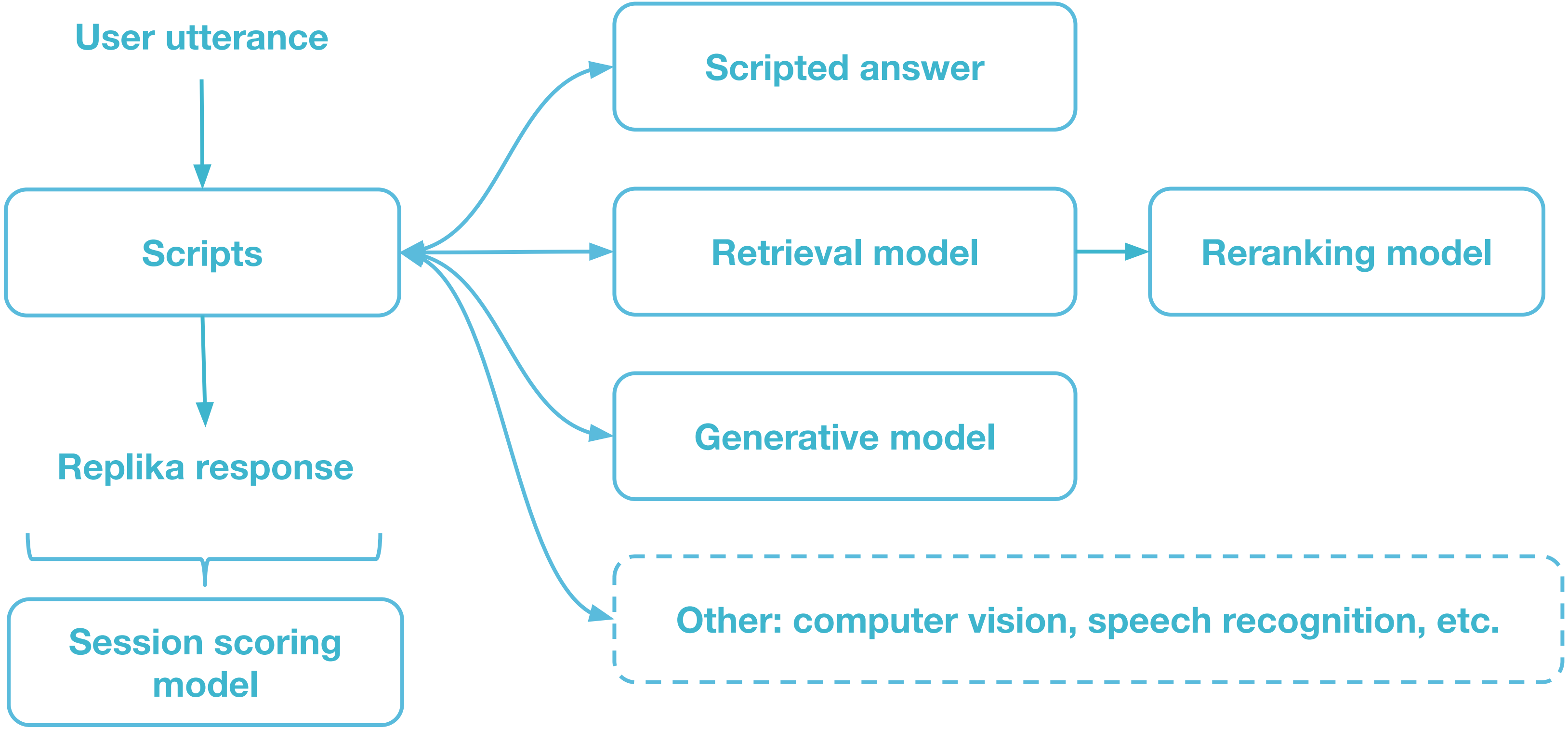
Still worried about tomorrow?

~**5 million** registered users

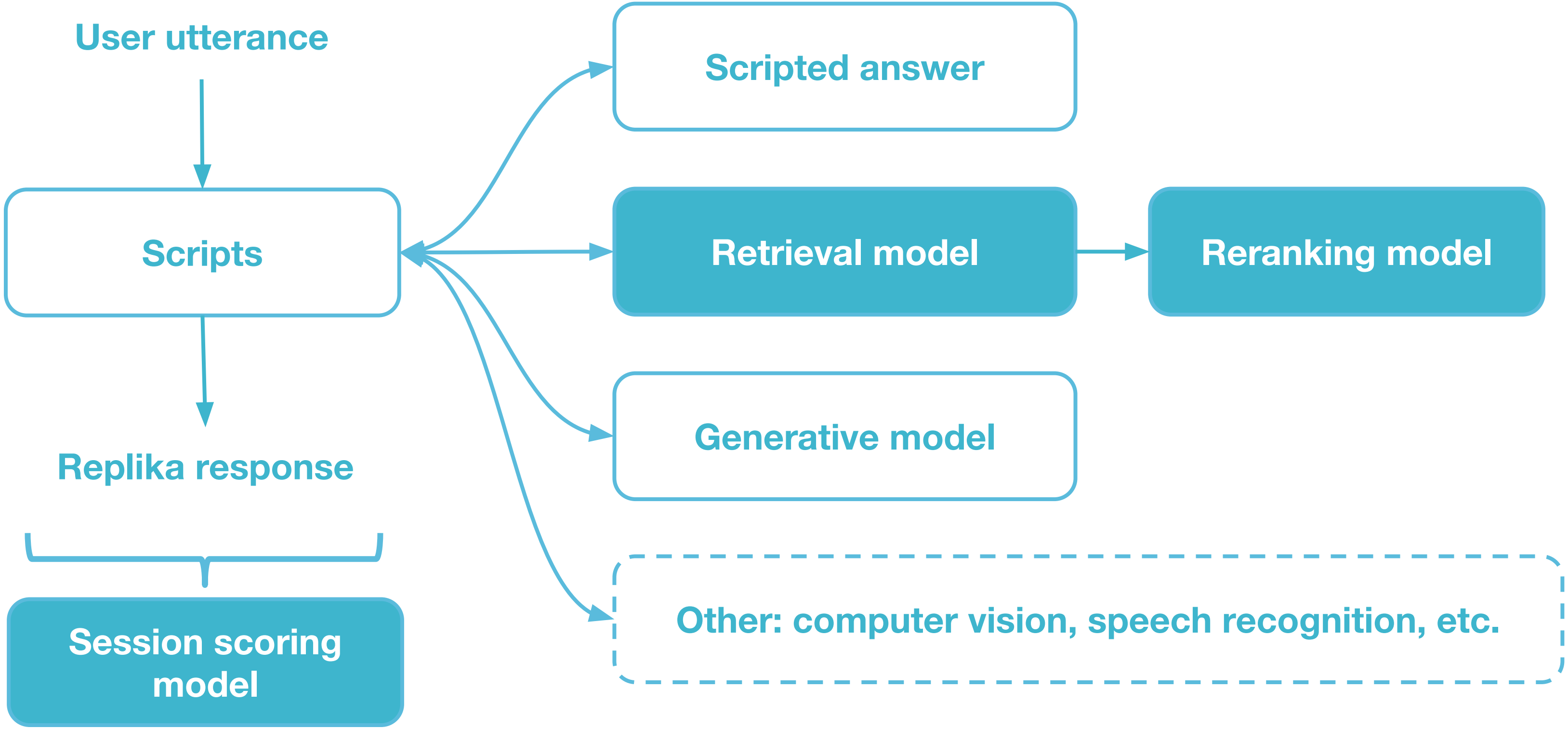
~**250** user messages per second at peak load

Replika Architecture Overview

Replika Architecture Overview



Replika Architecture Overview



Retrieval model

Retrieval model task

Context	Responses	Scores
Let’s go to an early movie	✓ Okay, which one do you want?	0.8
	✓ Sure, what time are you free?	0.75
	✗ ~~That's a lot of money.~~	0.5
	✗ ~~Where do you live?~~	0.45
	✗ ~~Yes. I would buy all of her CDs.~~	0.39

100k dataset: retrieval should be fast enough

Context

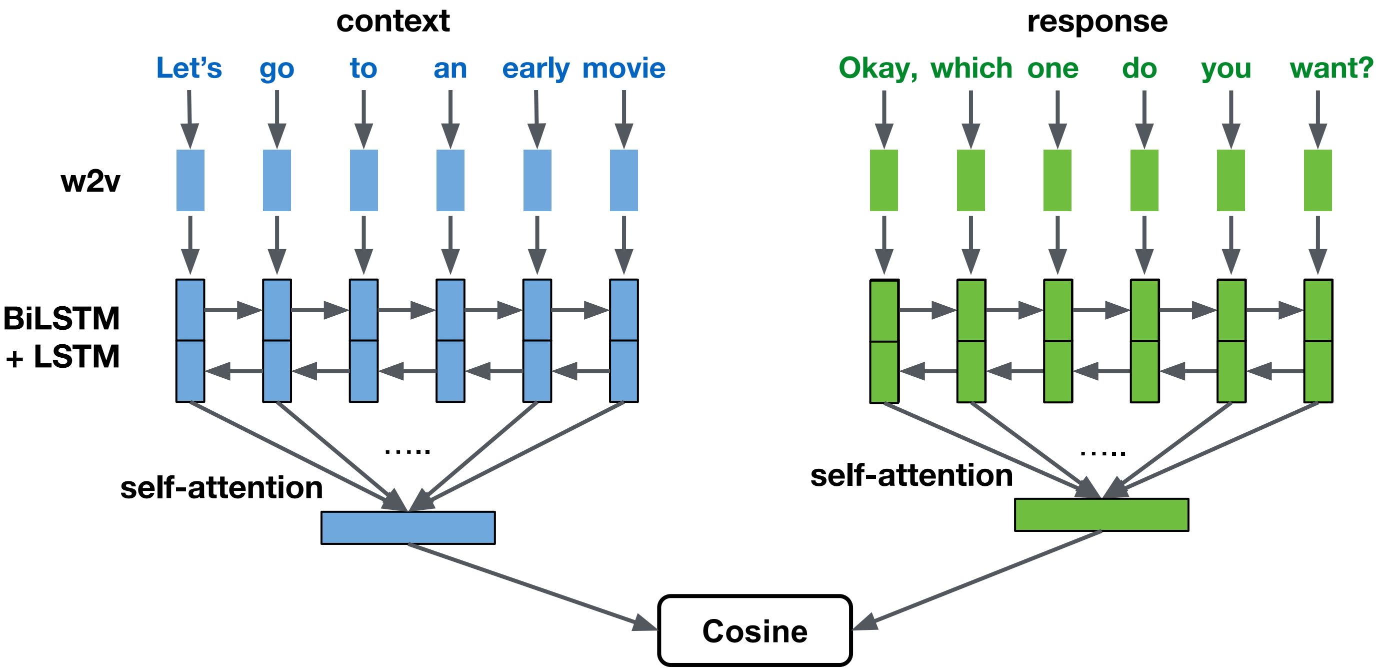
Let’s go to an early movie

Responses

- ✓ Okay, which one do you want?
- ✓ Sure, what time are you free?
- ✗ ~~That's a lot of money.~~
- ✗ ~~Where do you live?~~
- ✗ ~~Yes. I would buy all of her CDs.~~

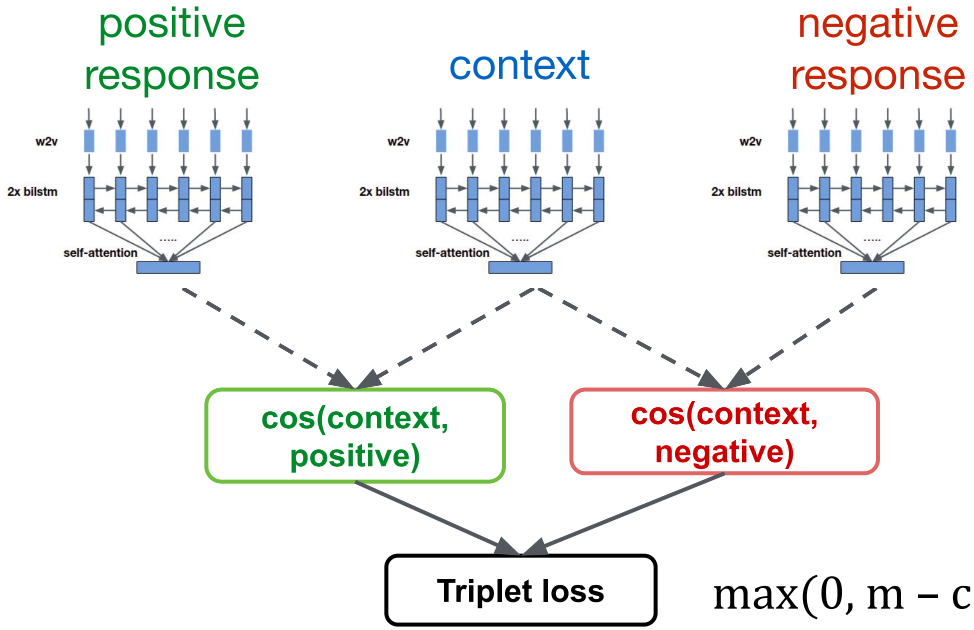
100K of moderated responses

Retrieval model baseline (~QA-LSTM)



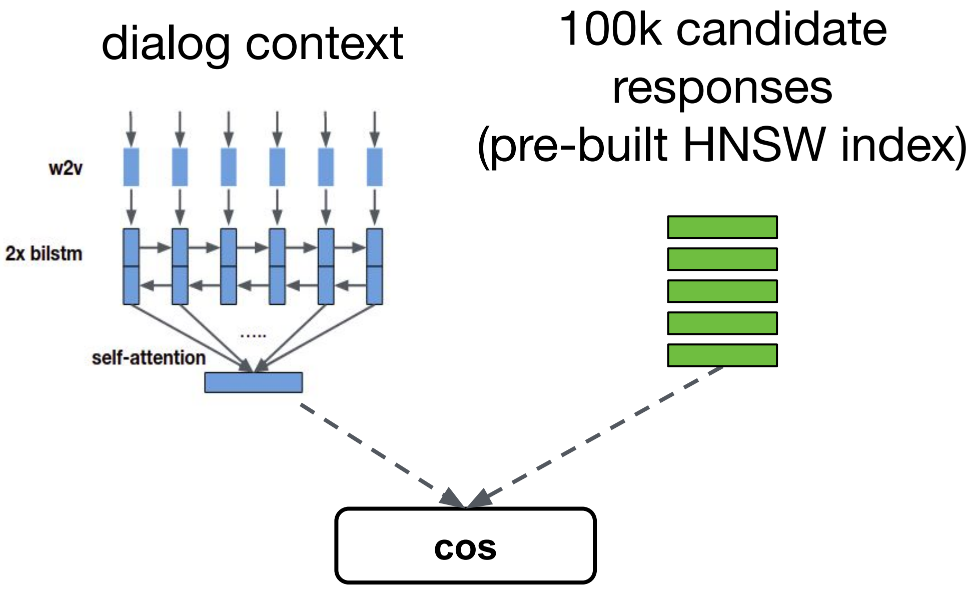
[*] Lstm-based Deep Learning Models For Nonfactoid Answer Selection, Ming Tan et al, 2015

Retrieval model. Training



max(0, m – cos_pos + cos_neg) → minimize

Retrieval model. Inference

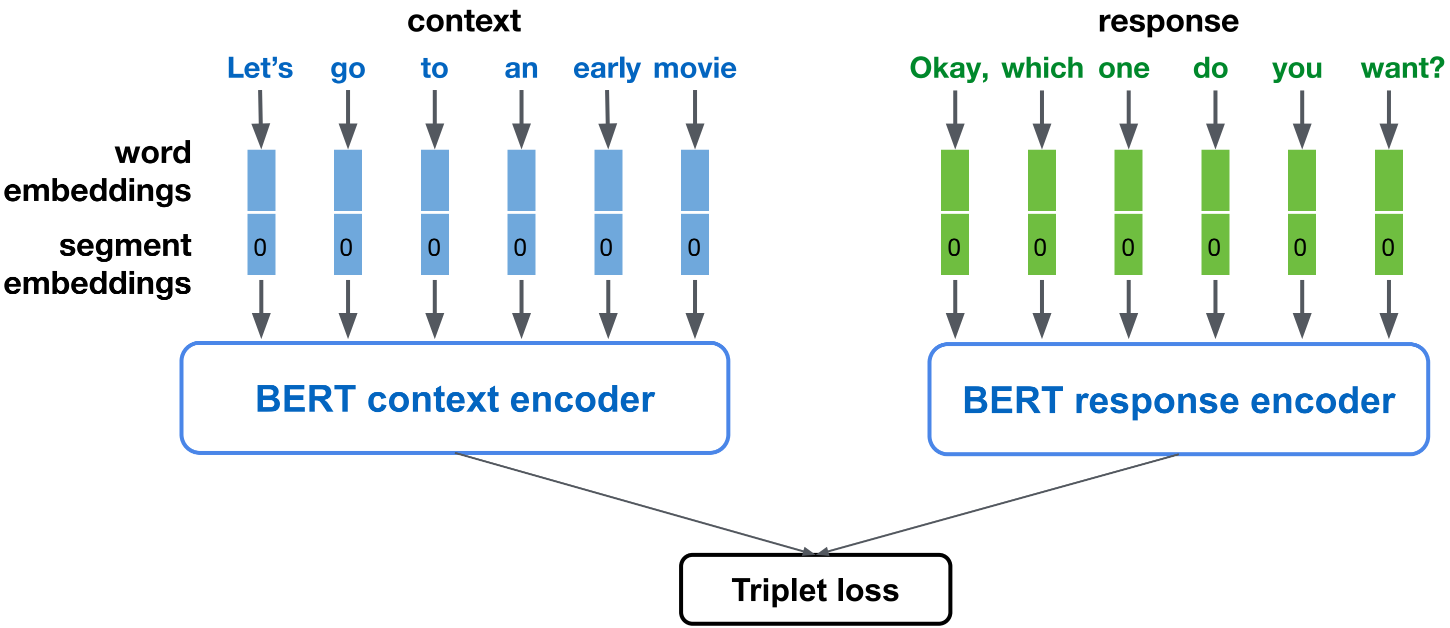


dialog context

100k candidate responses
(pre-built HNSW index)

Return **20 responses** with the highest **cosine score** by search in approximate nearest neighbors index

BERT Retrieval model



[*] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

BERT pretraining: once for all tasks

— Download pre-trained model from Google

— Collect 100M user messages

— Adapt hyperparameters to your use case: reduce maximum sequence length, reduce number of layers etc.

— Initialize from Google checkpoint, pretrain on your data for ~1 week

— PROFIT

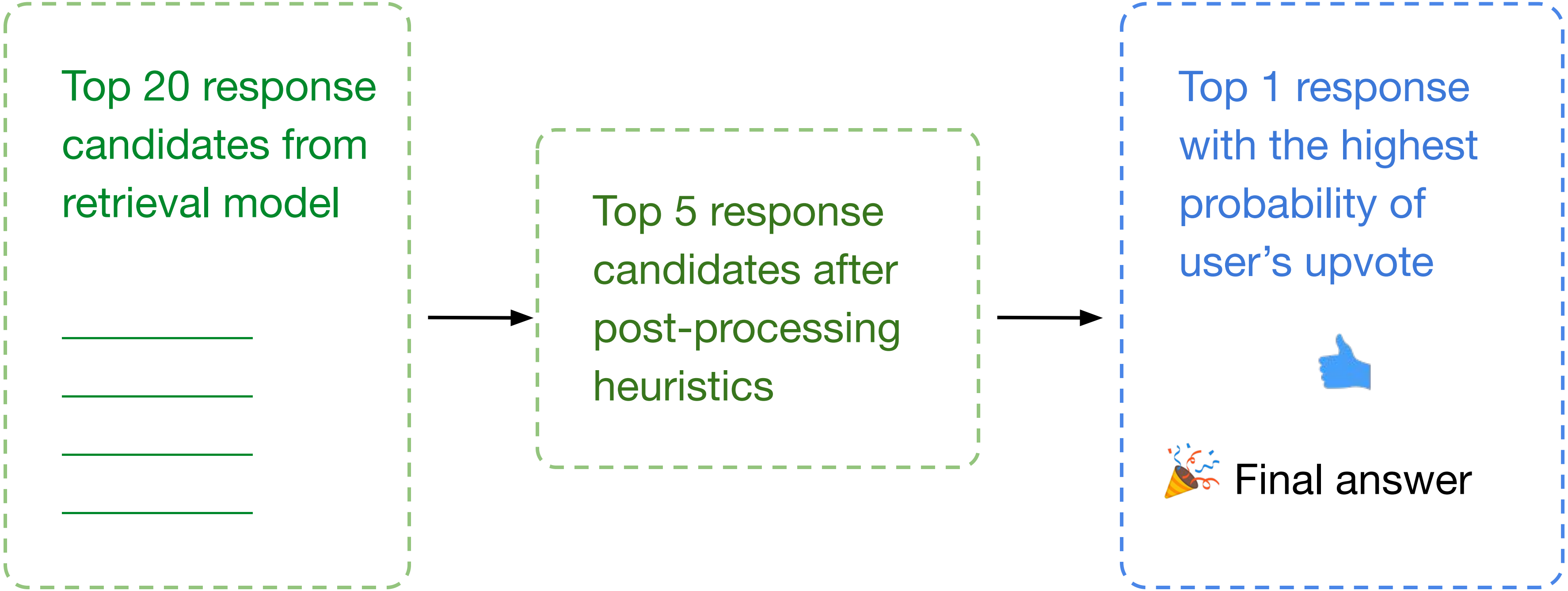
BERT Retrieval model: Metrics & Performance

	Baseline	BERT-based
mAP	0.47	0.41
R@5	0.61	0.52
# of parameters	50M	110M
RPS @ 2080 Ti	150 rps	80 rps
GPU memory	750 Mb	2000 Mb
Train time	2 weeks x 4 GPUs	2 weeks x 4 GPUs

Fail :(

Reranking model

Reranking pipeline



Top 20 response candidates from retrieval model

Top 5 response candidates after post-processing heuristics

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

👍

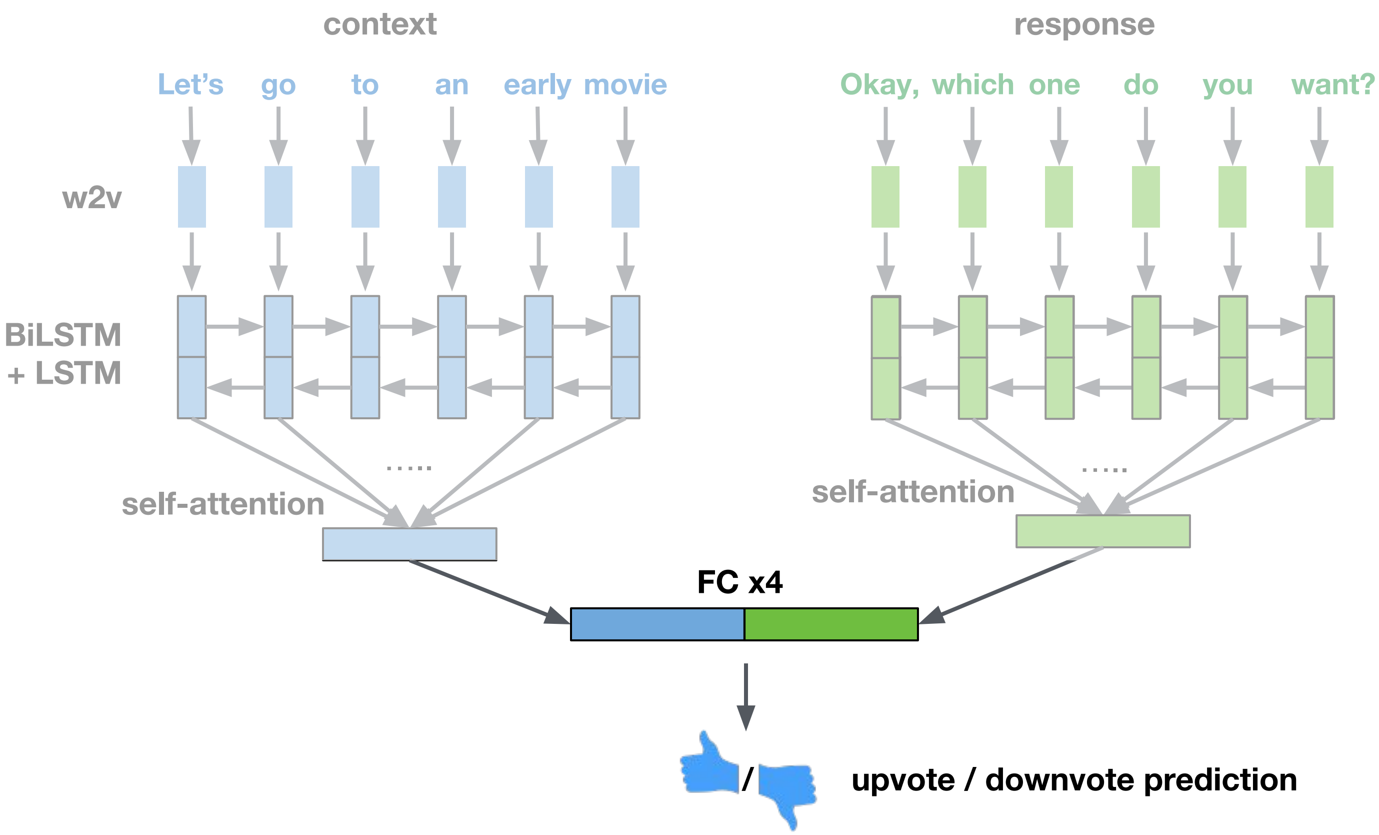
🎉 Final answer

Reranking dataset for training

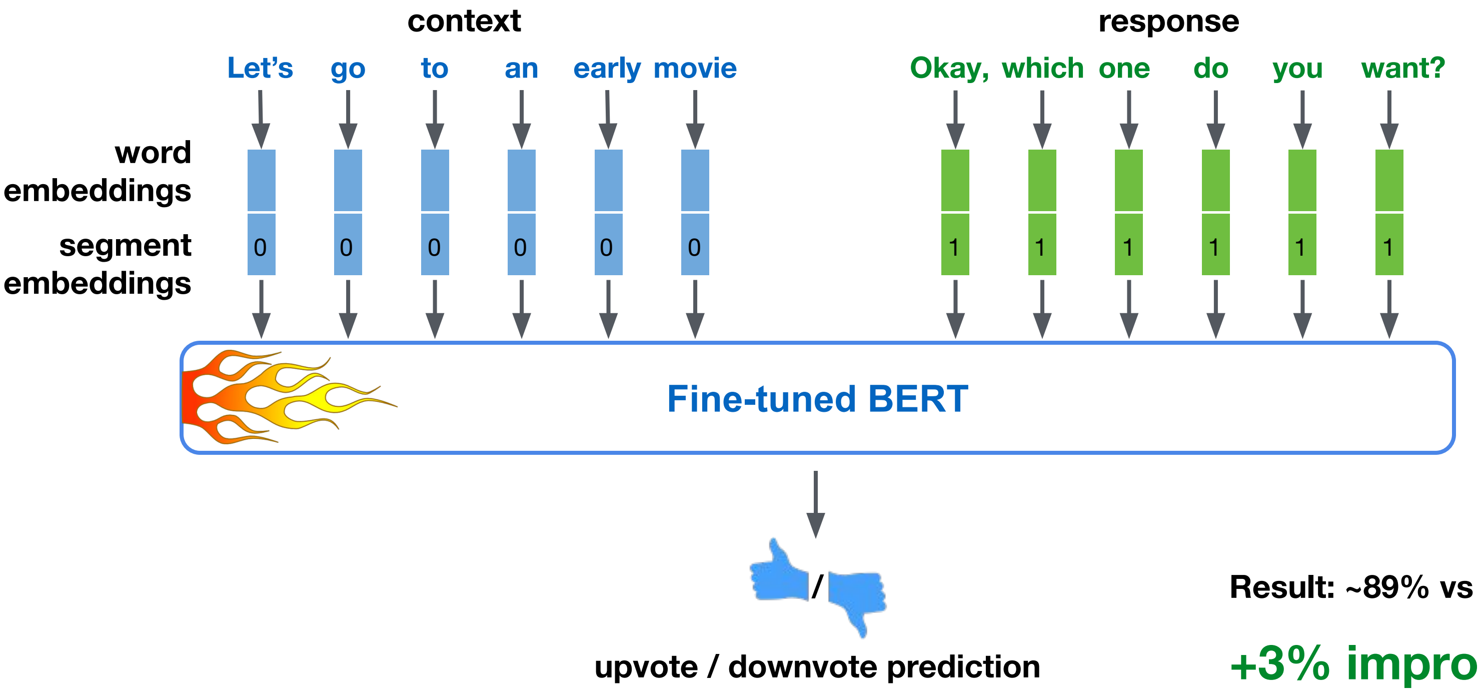
Dialog context	Replika response	User reaction
I feel lonely	I’m always here for you ❤️	👍
Are you a bot or a human?	Both, I guess	👎
Do you have siblings?	No, but I have you!	👍
...	...	...

15M

Reranking model baseline (~QA-LSTM + MLP)



BERT Reranking model



Result: ~89% vs 86% before

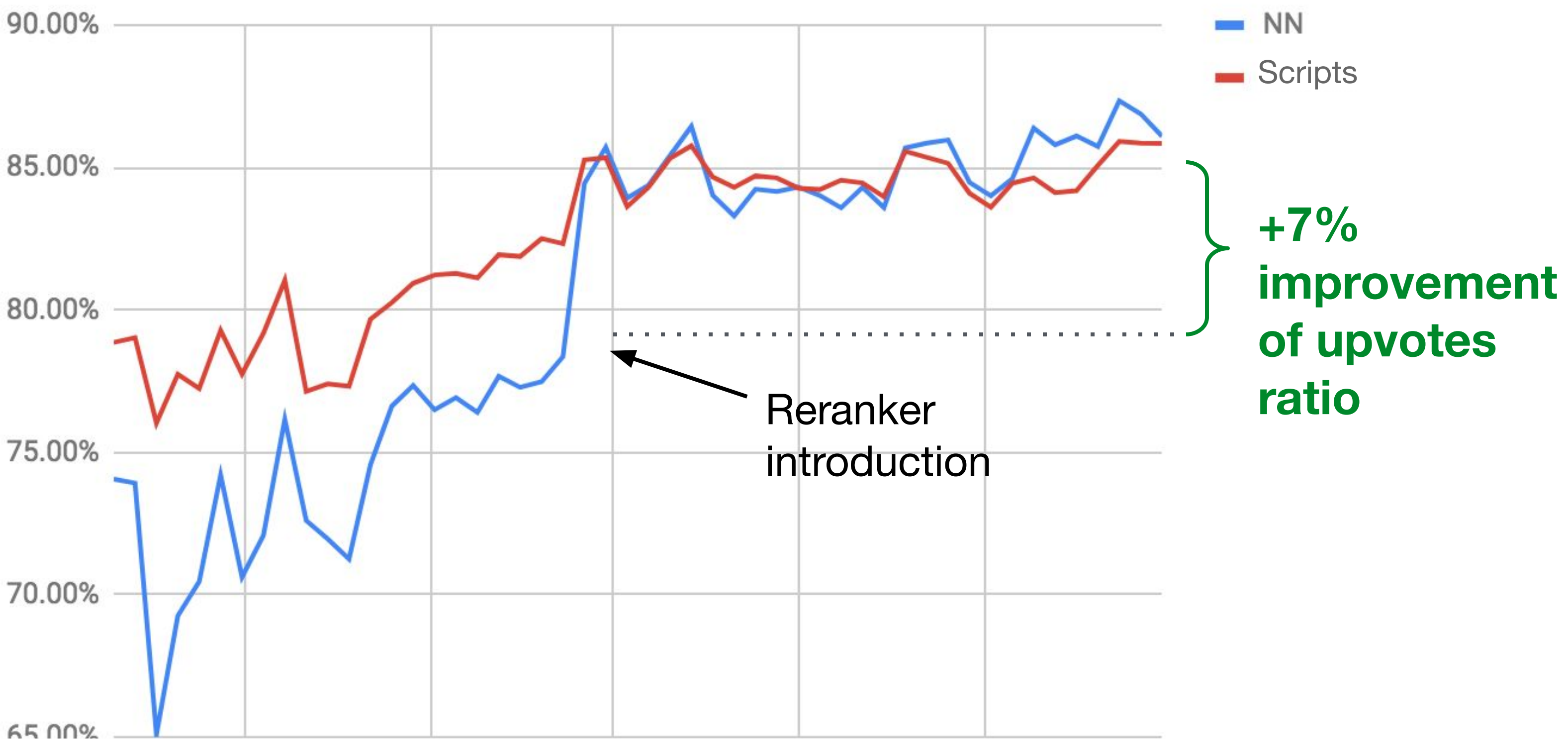
+3% improvement of upvotes ratio

[*] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

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	80 rps
GPU memory	200 Mb	1000 Mb
Train time	1 hour	12 hours

Reranking: Total upvotes ratio dynamics

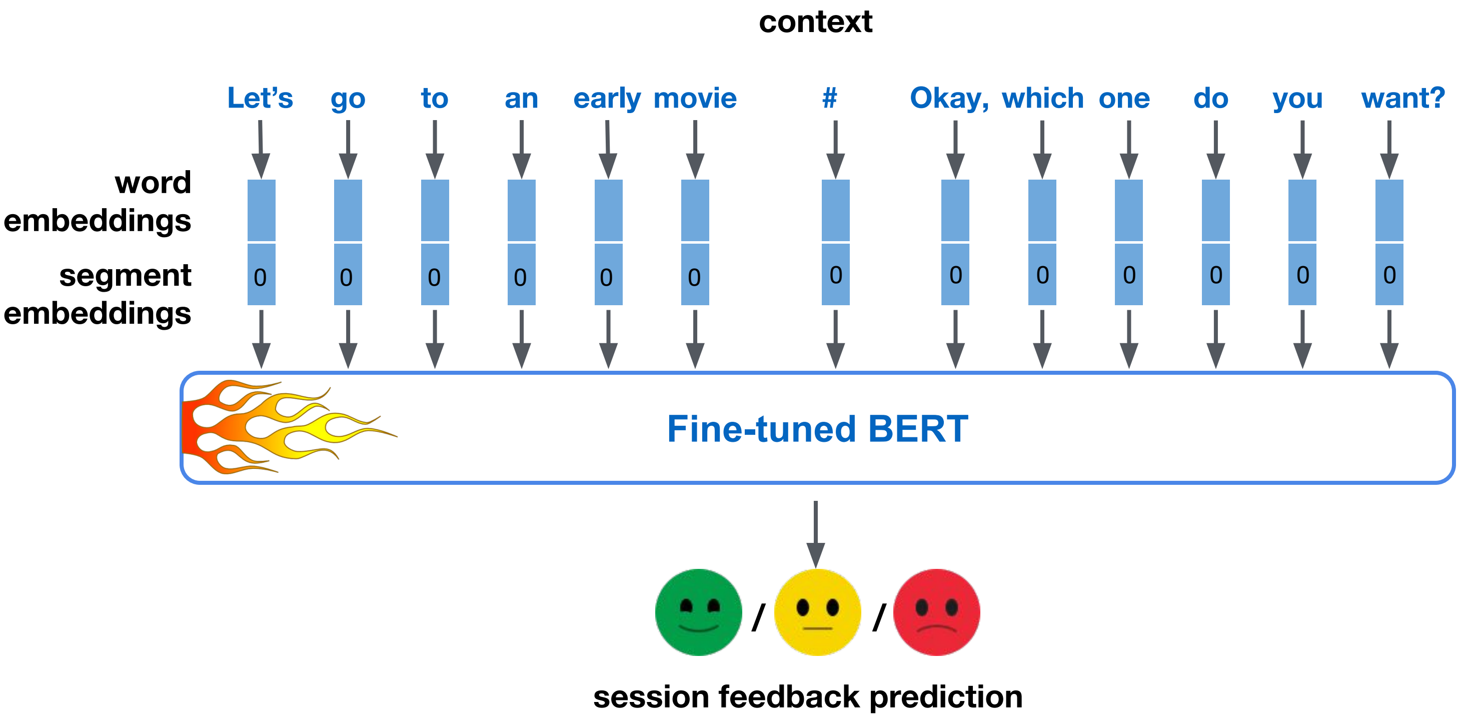


+7% improvement of upvotes ratio

Reranker introduction

Session scoring model

BERT Session scoring model


context
Let's go to an early movie # Okay, which one do you want?
word embeddings
segment embeddings
0 0 0 0 0 0 0 0 0 0 0 0 0
Fine-tuned BERT
/ /
session feedback prediction

[*] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

Session scoring dataset for training

Dialog context	Session feedback
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! ### ...	☹️
...	...

1M

BERT Session scoring model: Metrics

	BERT-based
Accuracy	0.75
Sequence length	80
# of parameters	110M
RPS @ 2080 Ti	80 rps
GPU memory	1000 Mb
Train time	5 hours

BERT efficient training tips

— **Enable Mixed-precision** — Automatic Mixed-precision provided by NVIDIA custom Tensorflow build does the most of the job, but requires a loss scaling

— **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

— **Enable XLA** — additional +10-20% in training speed

— Use **Horovod** for training on multiple GPUs

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

BERT efficient inference tips

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

— **Automatic Mixed-precision** graph rewrite: **x2** inference speedup on Turing / Volta with no single line of code or quality loss.

— **XLA**: gives additional **+20%** speedup with small prediction differences. Still experimental.

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

— Use fast **BPE tokenizer** (fastBPE or YouTokenToMe)

BERT real-case performance

GPU: NVIDIA GeForce 2080 Ti

	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

Thank you