SPLADE

a Sparse BERT model for Neural Information Retrieval

Thibault Formal, Benjamin Piwowarski (Sorbonne University), Carlos Lassance, Arnaud Sors,
Stephane Clinchant

NAVER LABS

Information Retrieval (IR) Models

DEVIEW 2021

NAVER

50's Boolean

70's Vectorial

90's Probabilistic

2000's Learning to Rank

2019- Pretrained LMs

Longstanding debate

DEVIEW 2021

Dense Model

$\mathbb{R}^{768}$

- Semantic
- Implicit Matching
- 'Representation Based'
- Approximate NN Search

Sparse Model

$\mathbb{R}^{30k-500k}$

- Exact Match
- Explicit Matching
- 'Interaction Based'
- Inverted Index

Now Dense > Sparse

How can one learn a state of the art sparse retrieval model?

SPLADE

A spork that is sharp along one edge, or both edges, enabling it to be used as a knife, a fork and a spoon.

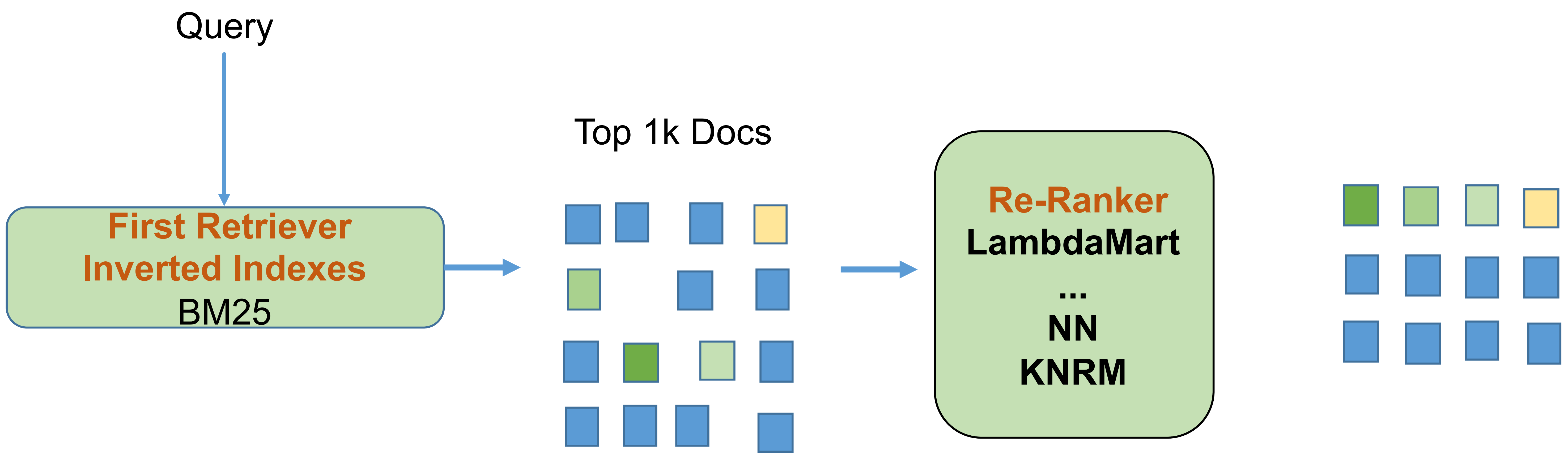
CONTENTS

Sparse Lexical AnD Expansion Model for First Stage Retrieval

The first Sparse Model to rival Dense Models

1. An introduction to Neural Information Retrieval

Anatomy of a Search Engine



BM25, Robertson et al., 1994

Hypothesis:

word frequencies follow a two Poisson Mixture

$$\sum_{w\ in\ q \wedge d} \frac{tf(w)}{tf(w) + K} IDF(w)$$

The backbone of search engines for several decades

Classical Rerankers

Rerankers: Learning-to-rank methods :

- LambdaMart, RankNET, GBDT on handcrafted features

2010's: NN models with word embedding (word2vec)

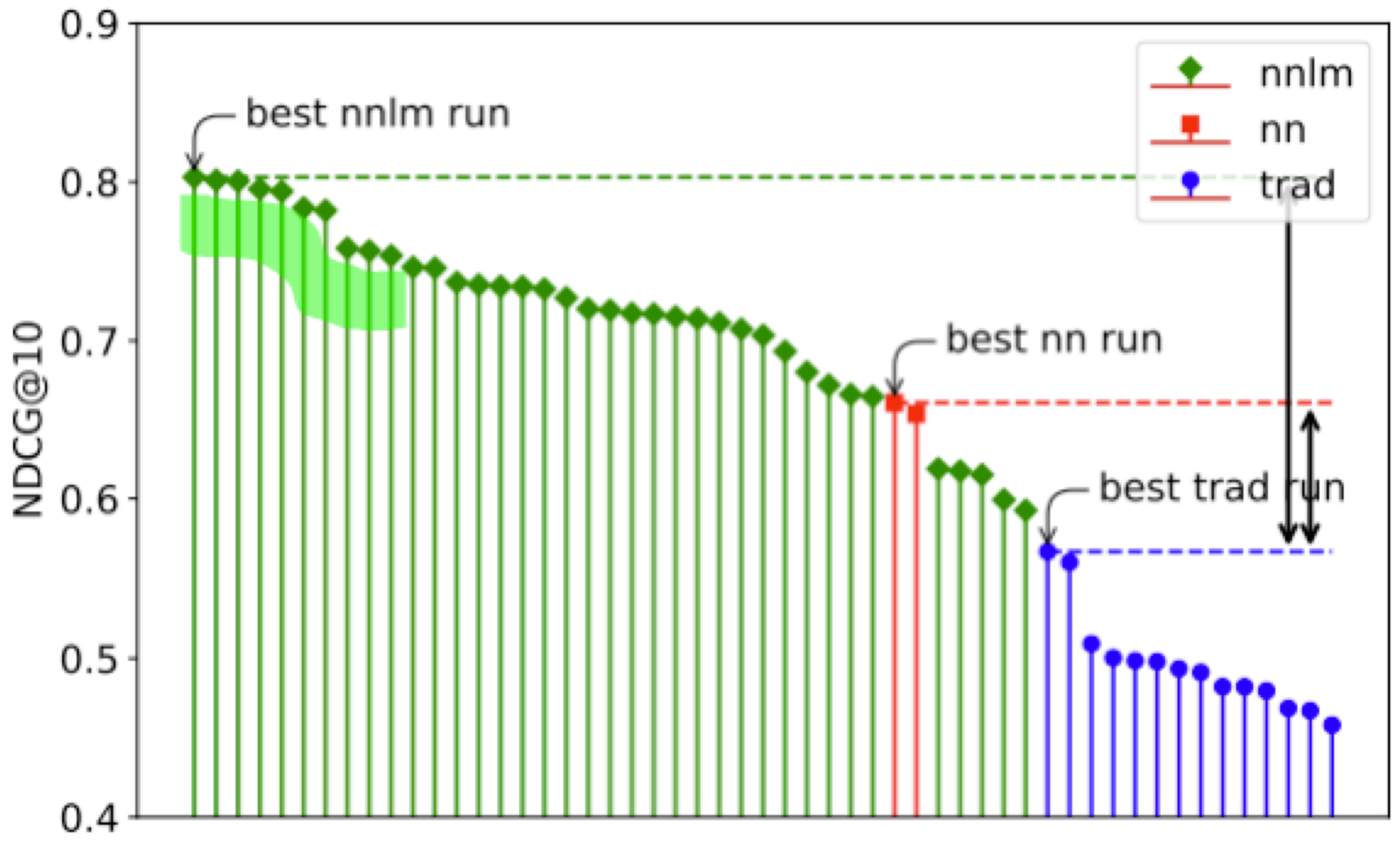
- Representation based e.g. DSSM
- Interaction based e.g. DRMM, K-NRM, DUET

MSMARCO and TREC

Information Retrieval Competition since 90's

2019

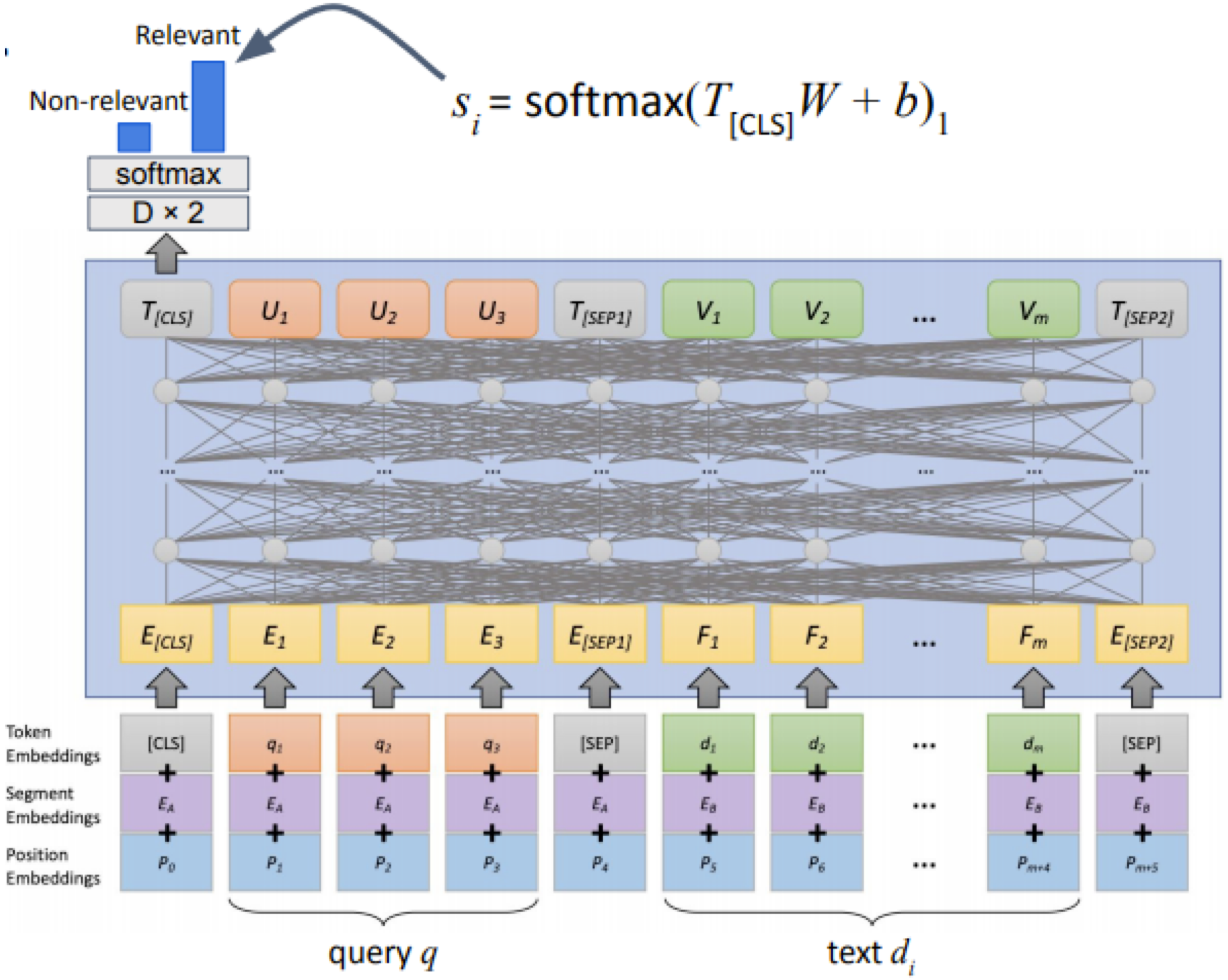
Bert and Transformers



Huge Gain but High Computational Cost

BERT Reranker: BERT (Cat)

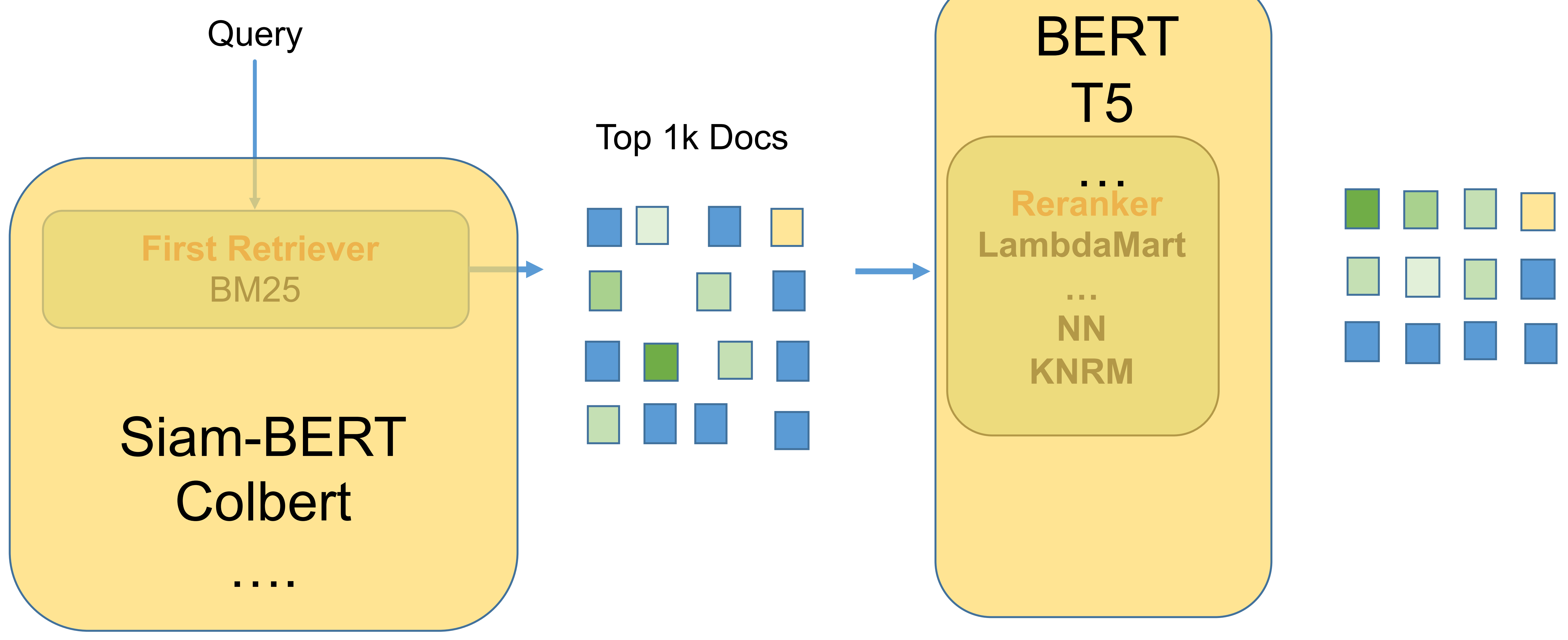
DEVIEW 2021



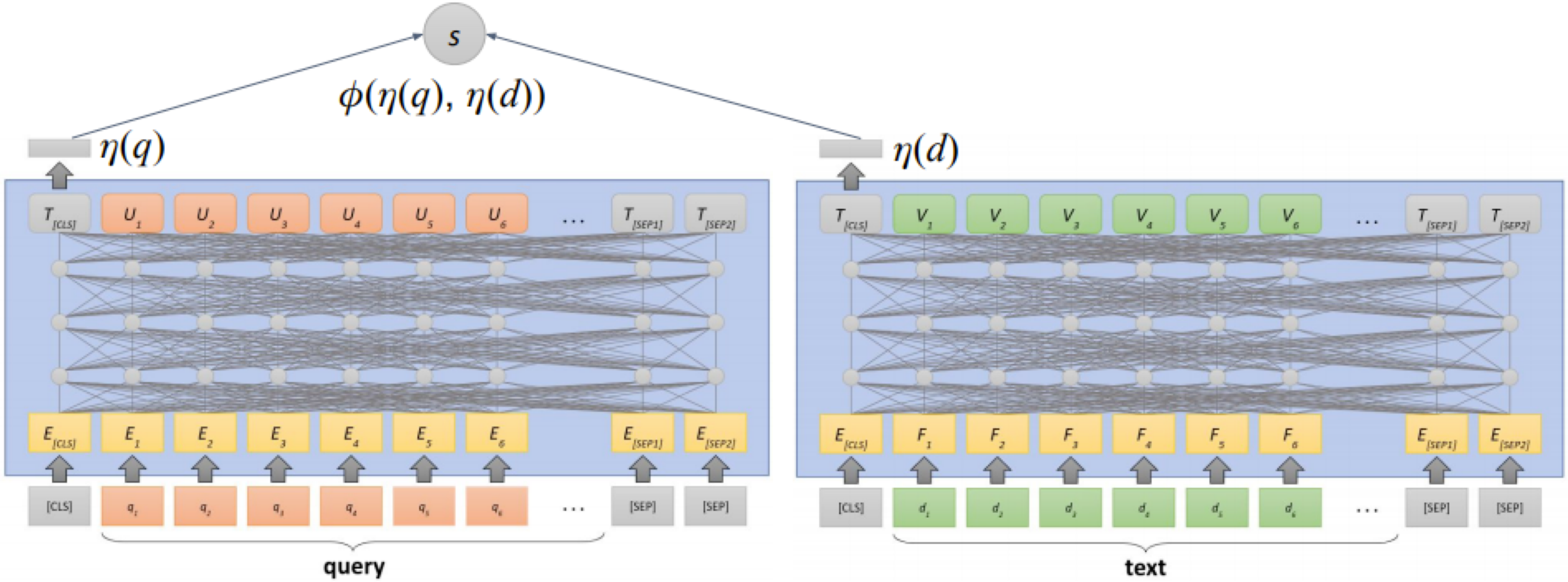
FT with various learning to rank loss *on the Top1k documents returned by BM25*

Schema credit: Lin Nogueira, Yates in *Pretrained Transformers for Text Ranking: BERT and Beyond*

Pretrained LMs for First Retriever and Rerankers



A Bi-Encoder First Stage Ranker



From Inverted index to dense indexing technique (ANN)

Schema credit: Lin Nogueira, Yates in
Pretrained Transformers for Text Ranking: BERT and Beyond

First Ranker Comparison: MS-Marco and TRECDL'19

Model	MRR@10 MSMARCO Dev	NDCG@10 TREC DL19
BM25	19.4	50.1
docT5	27.7	64.2
Siamese Bert	31.2	63.7
TAS-B	34.7	71.7

Research Questions

DEVIEW 2021

How to reduce computational cost

e.g. quantization, distillation of reranker to a siamese

How better train these models

e.g. multi-stage training, label noise

Generalization?

Archi-tecture

Distillation

Ranking Loss

Multi-Stage Training

BEIR Benchmark: Zero Shot Evaluation, Neurips’21

DEVIEW 2021

BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, Iryna Gurevych
Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Department of Computer Science, Technische Universität Darmstadt
www.ukp.tu-darmstadt.de

Figure 1: An overview of the diverse tasks and datasets present in BEIR.

BEIR Benchmark: Zero Shot Evaluation, Neurips'21

BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, Iryna Gurevych
Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Department of Computer Science, Technische Universität Darmstadt
www.ukp.tu-darmstadt.de

Pause a moment

What's your bet of this benchmark ?

BEIR Conclusion

BM25	Colbert	TAS-B
45.3	45.6	43.7

- Rerankers transfer well
- Standard siamese don't
- Colbert ok too

"Our results show BM25 is a robust baseline … In contrast, Dense-retrieval models […] often underperform other approaches, highlighting the considerable room for improvement in their generalization capabilities "

2. A White Box Analysis of Colbert

And the important role of Exact Match - that will guide us to the design of SPLADE

A Research Question

DEVIEW 2021

IR Theory

IDF Interpretation

Axiomatic Methods

Relevance Estimation

….

Pretrained LMs

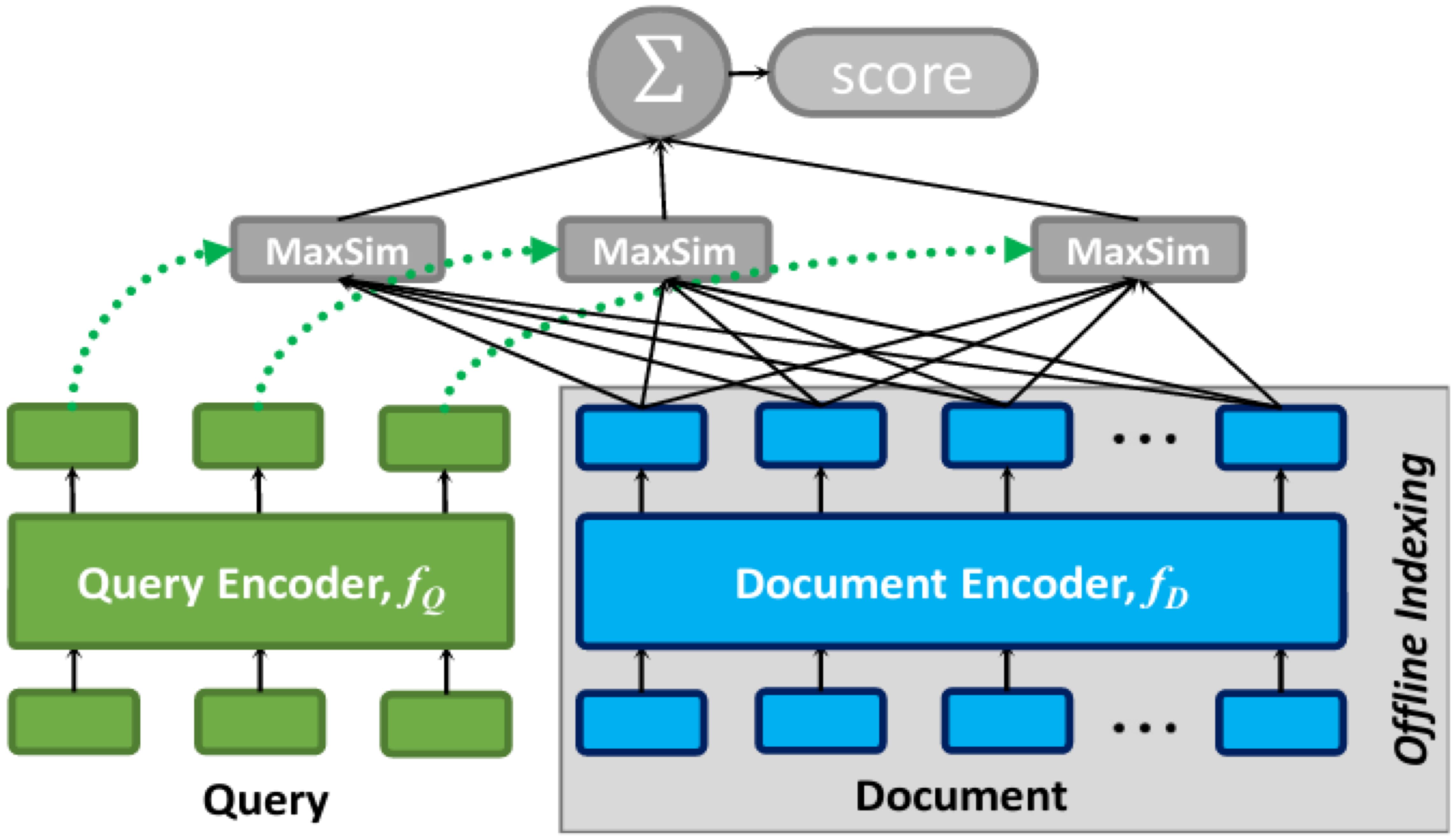
Work Better

What do they do?

Is IR Theory still useful?

ColBERT (SIGIR20, Katthab et al.)

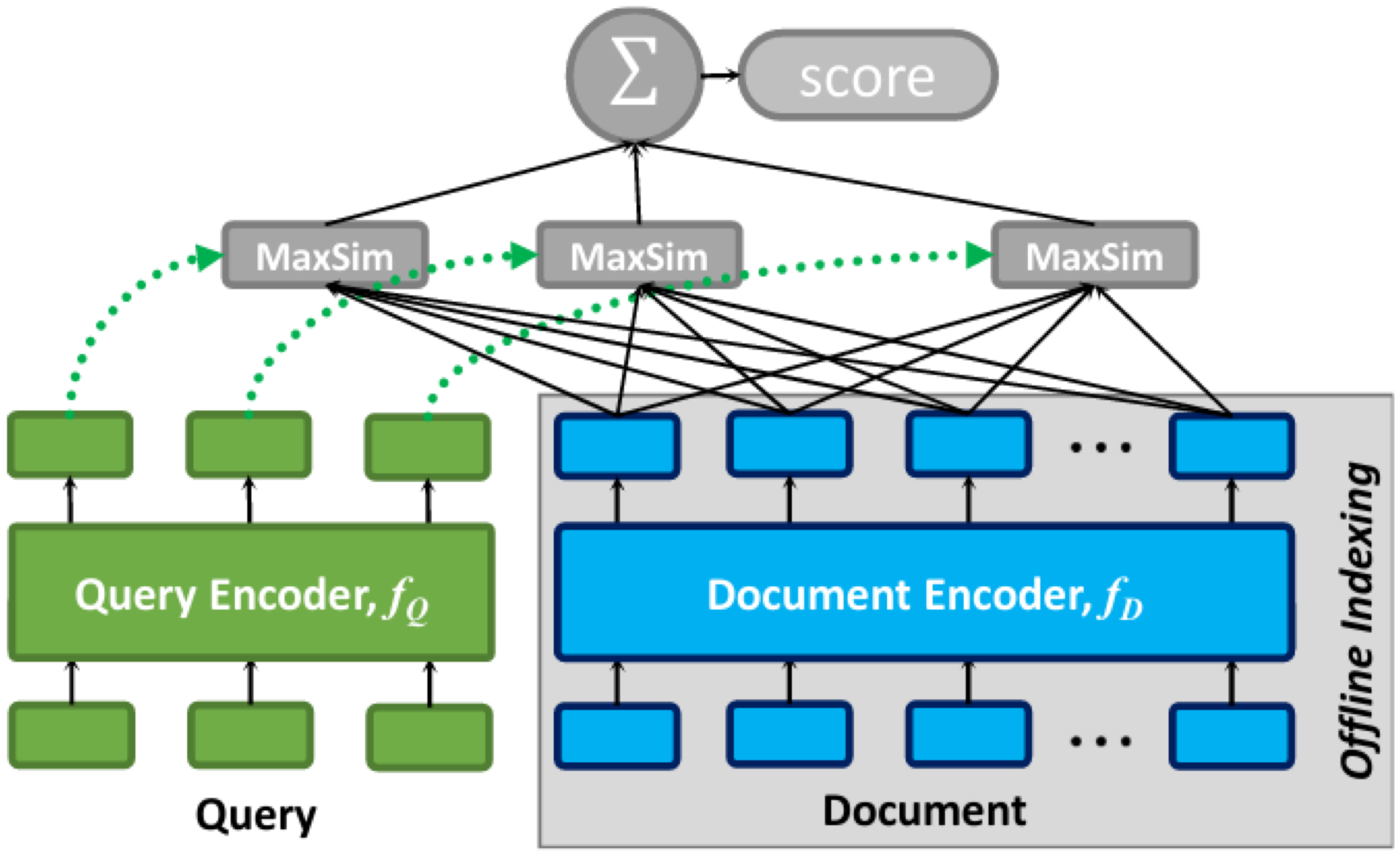
Delayed token-level interactions between query and doc (offline doc indexing)



$$S_{q,d} := \sum_{i \in [|E_q|]} \max_{j \in [|E_d|]} E_{q_i} \cdot E_{d_j}^T$$

Works surprisingly well! Resembles a TFIDF-like formula

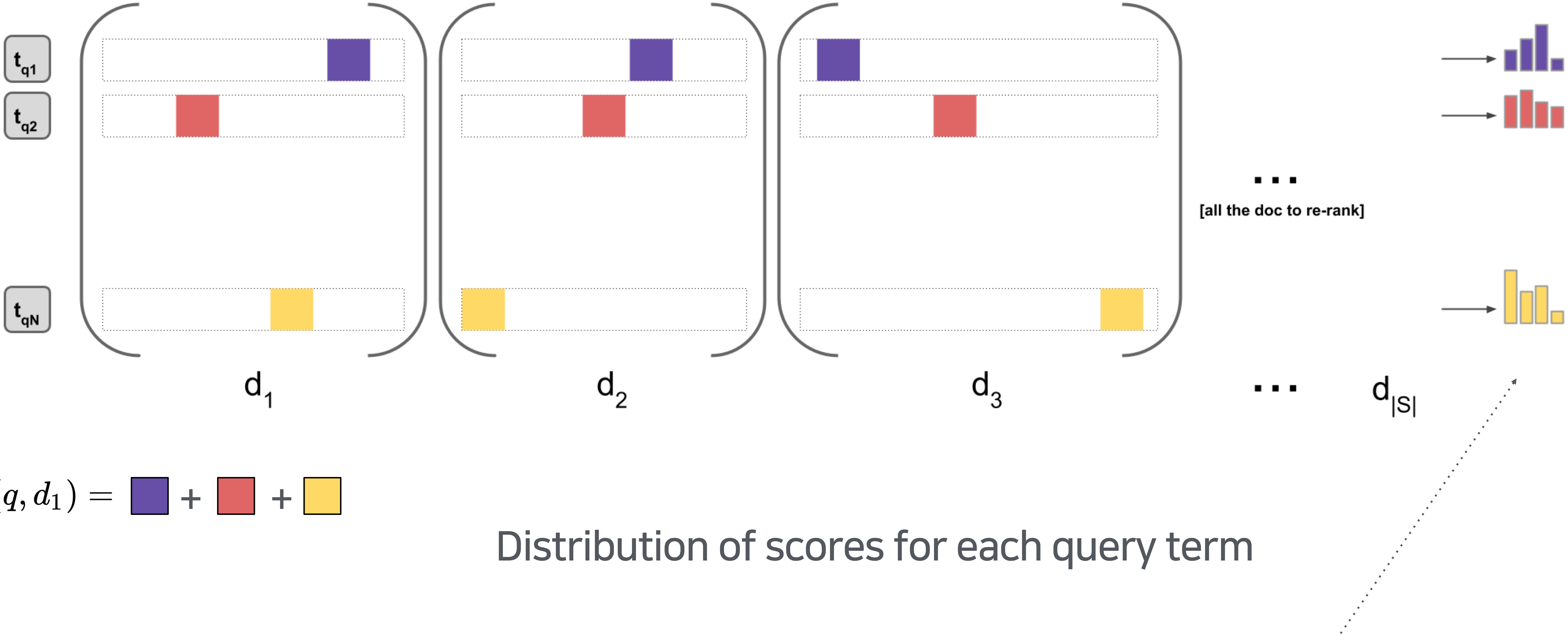
ColBERT Matching Process



$$S_{q,d} := \sum_{i \in [|E_q|]} \max_{j \in [|E_d|]} E_{q_i} \cdot E_{d_j}^T$$

- Statistics of scores for different terms on MS-MARCO
- Exact & Soft matches

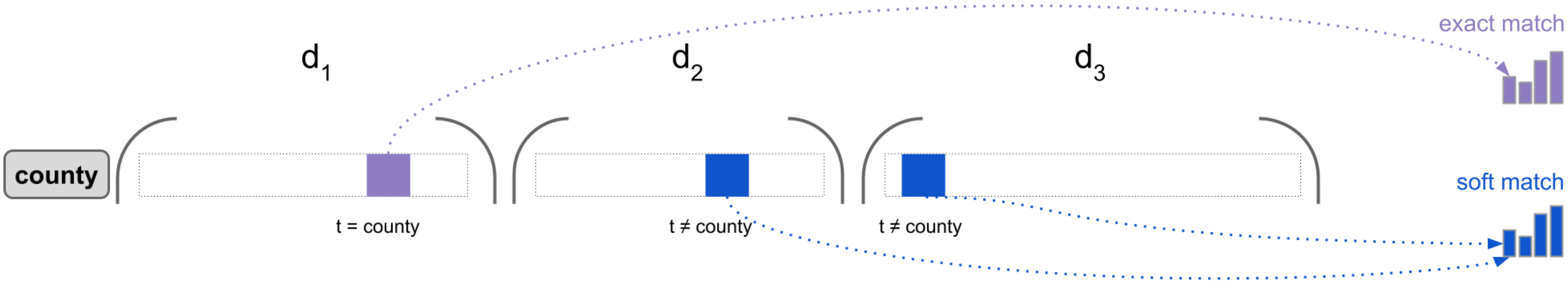
Methodology - distribution of term scores



$s(q, d_1) =$ ■ + ■ + ■

Distribution of scores for each query term

Methodology - exact and soft distributions

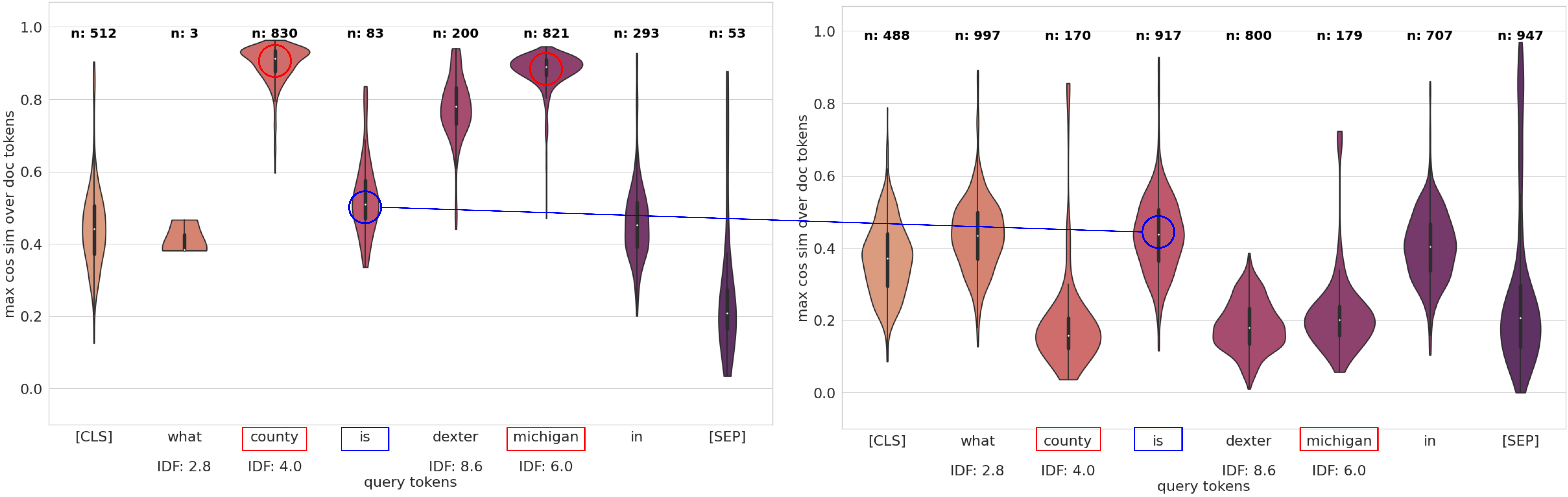


2 distributions of scores for each query term

- exact case
- soft case

Note ~ exact cosine sim != 1 because embeddings are contextualized

Motivation



exact match

soft match

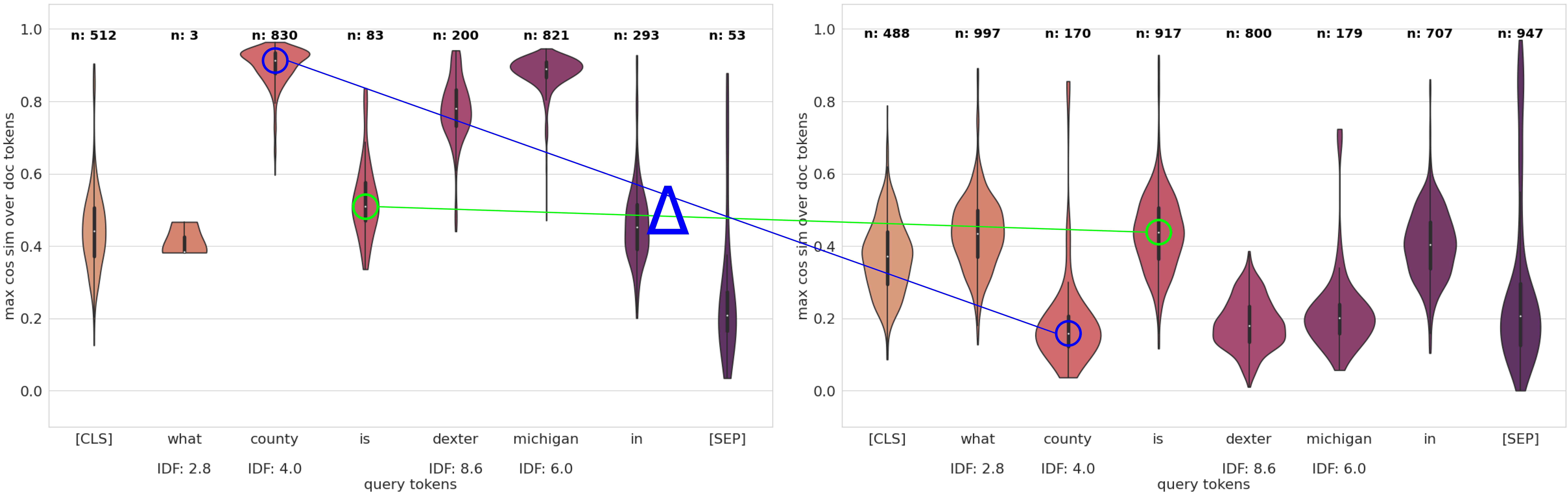
Exact/Soft matching patterns

Neural models ↝ soft-matching

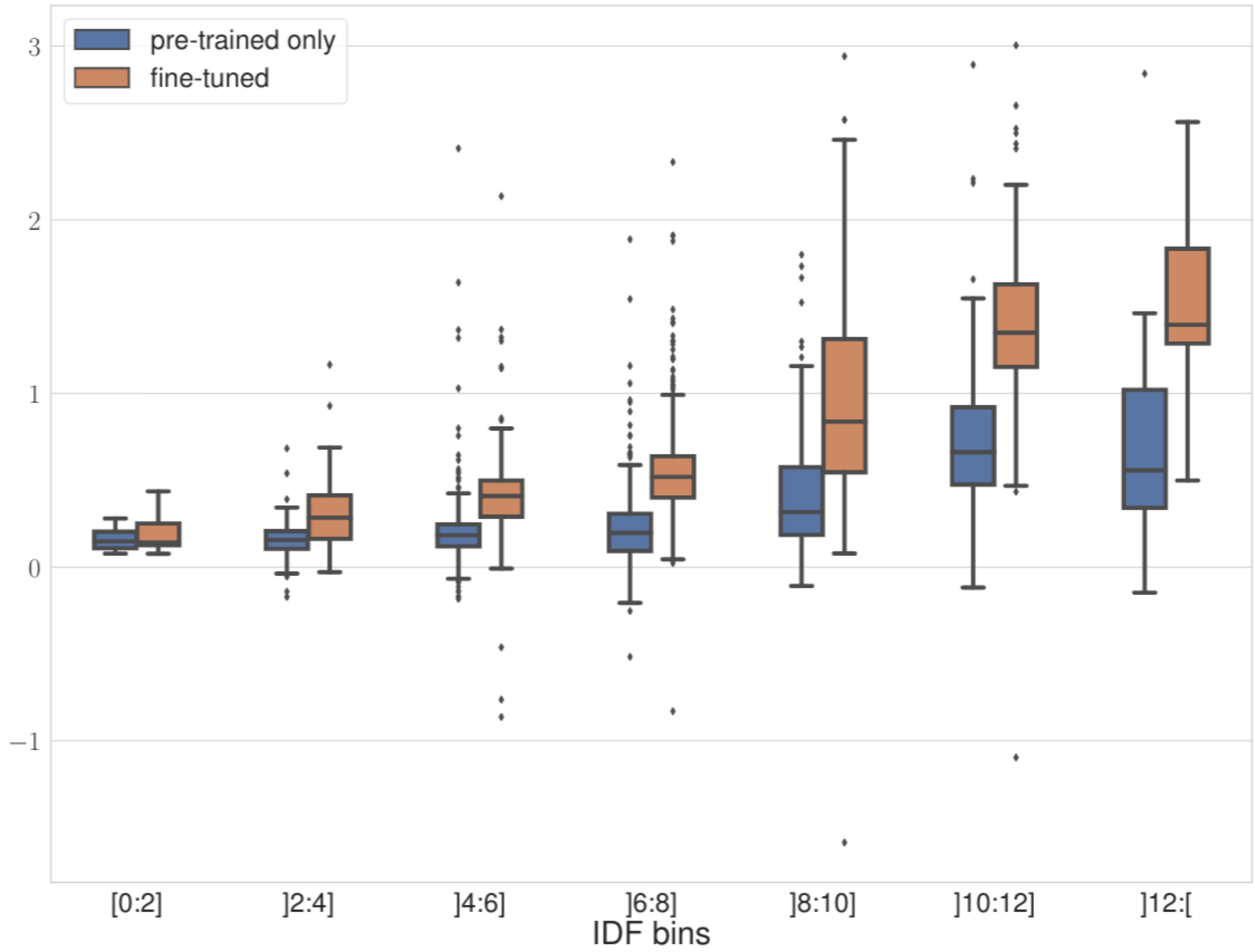
Exact matching is still a critical component of IR systems!

Does ColBERT capture exact match? How?

Exact/Soft matching patterns: Δ



2 - Exact/Soft matching patterns



Pearson r = 0.667

Exact Match: How ?

Colbert can distinguish terms for which exact match is important !

But how is it able to promote exact match from the contextualized embeddings ?

Exact Match in ColBERT: How ?

DEVIEW 2021

$$s(q,d) = \sum_{i \in q} \max_{j \in d} E_{q_i}^T E_{d_j}$$

Hypothesis

- for important terms, contextual embeddings vary less, hence ColBERT will tend to select the same term in documents (*cosine sim close to 1*)
- terms carrying less information tend to absorb more the context in sequences, hence their embeddings vary more

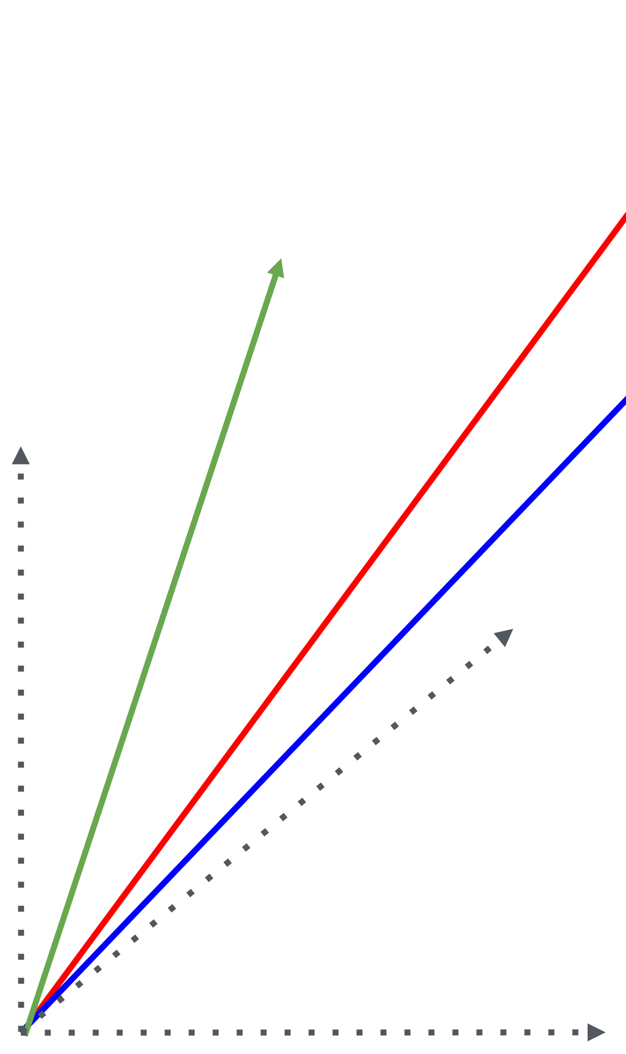
Hypothesis: content words have contextualized embeddings pointing in the same direction

[...] mango is an exotic fruit [...]

[...] mango is now cultivated in most frost-free tropical *[...]*

...

bla bla bla is mango



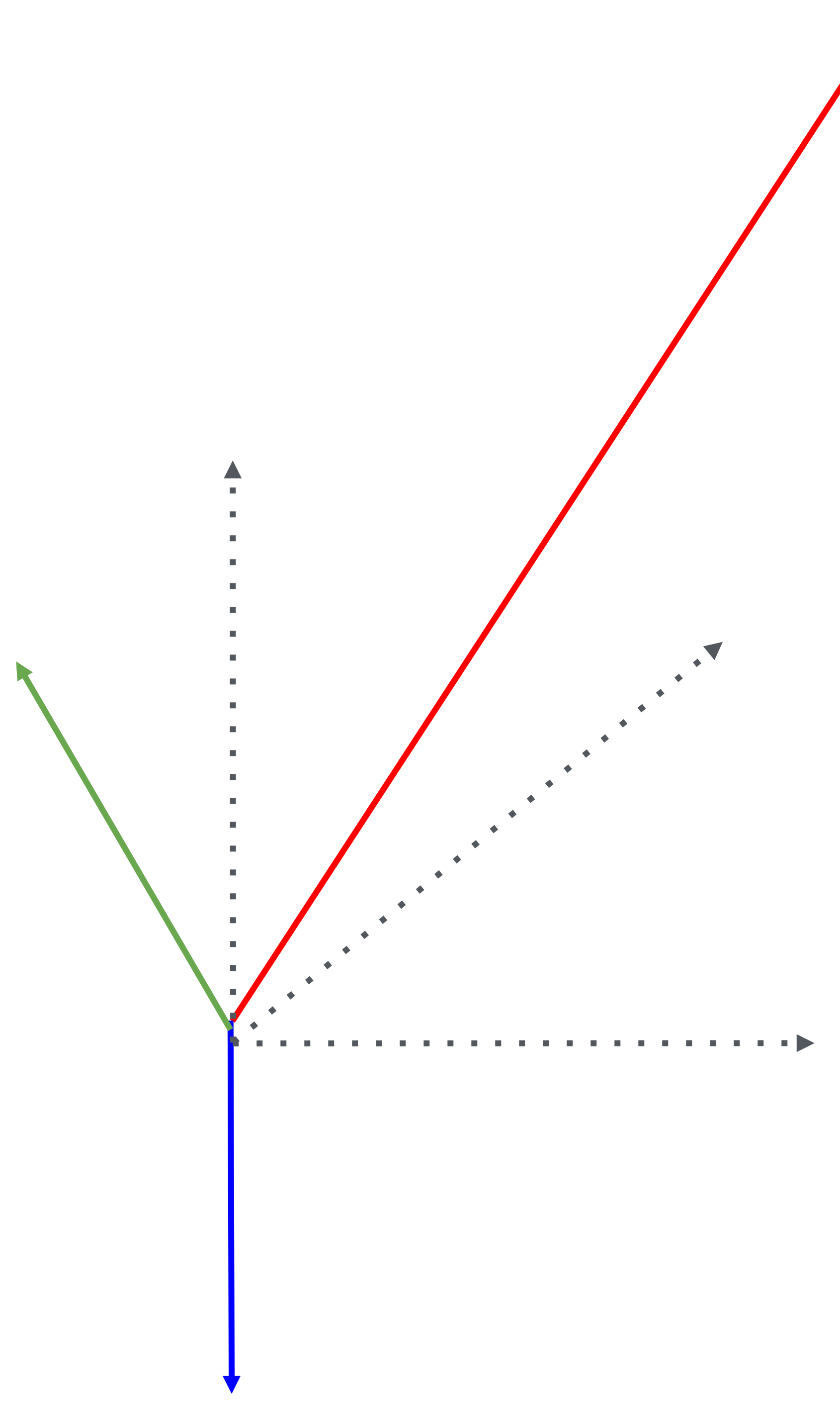
Hypothesis: frequent words have contextualized embeddings pointing in different directions

[...] mango is an exotic fruit
[...]

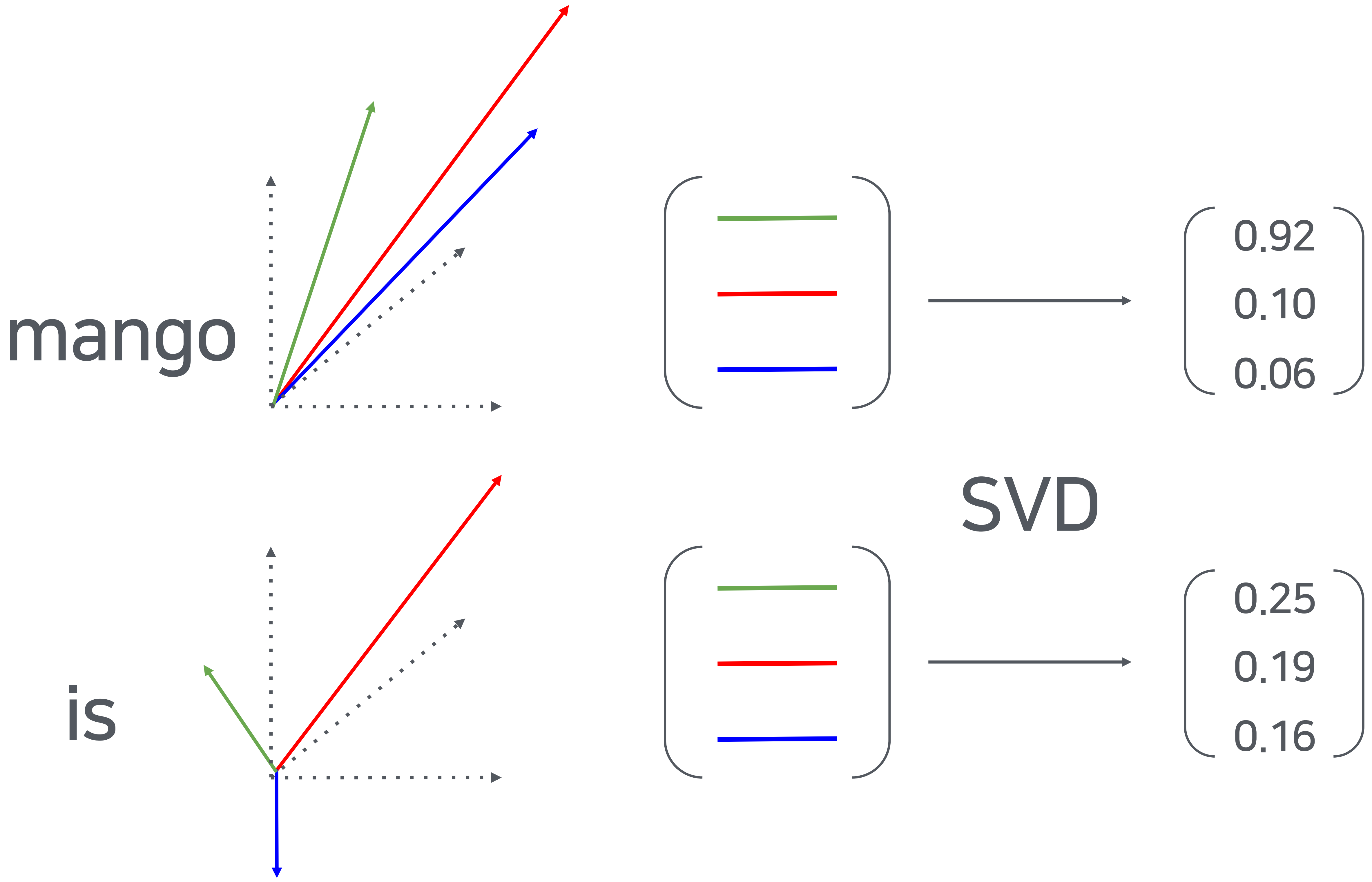
[...] mango is now cultivated in most frost-free tropical [...]

...

Bla bla is bla



Spectral analysis of contextual term embeddings

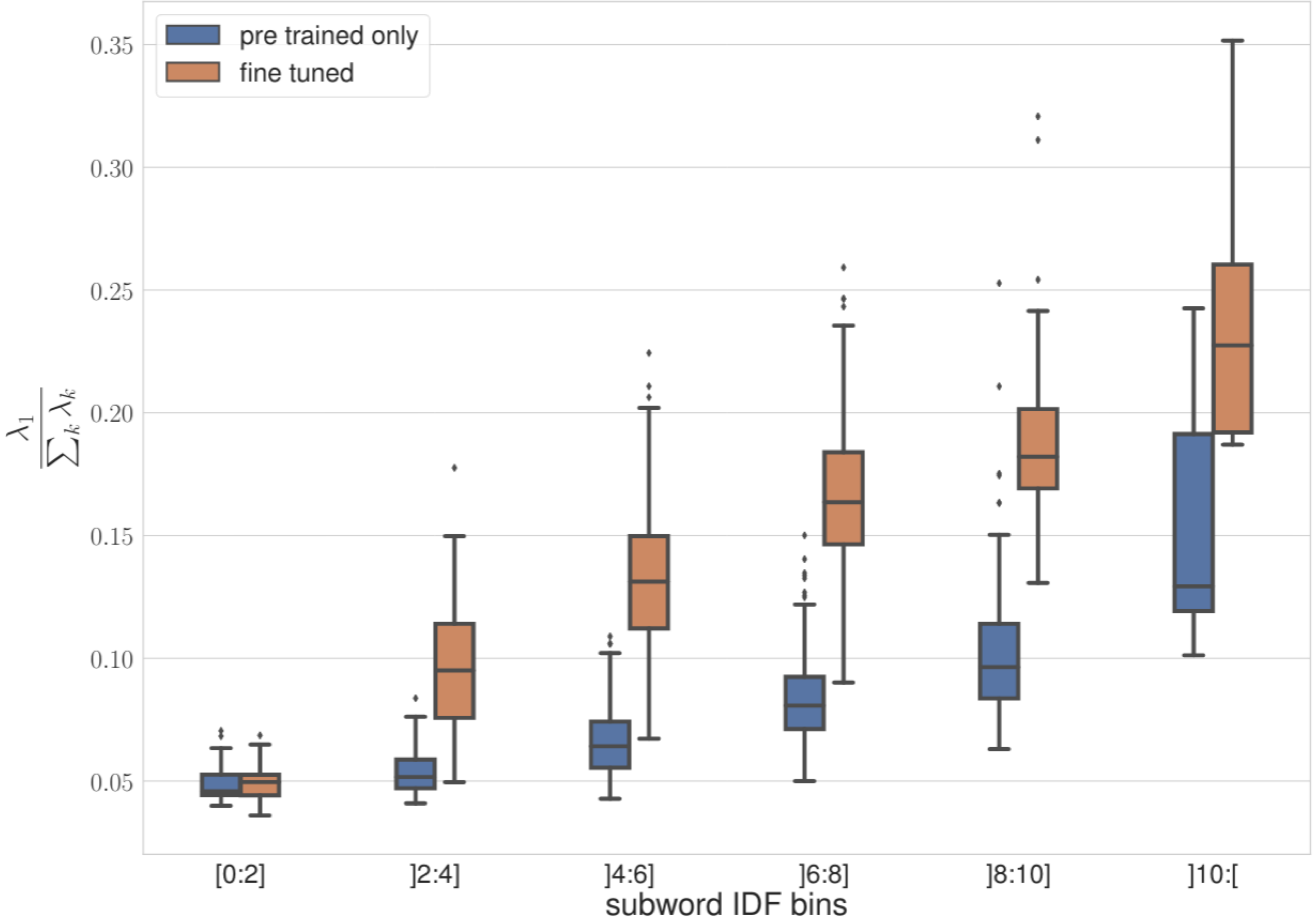


$$\frac{\lambda_1}{\sum \lambda_i}$$

High value means that embeddings point in the same direction

Spectral analysis of contextual term embeddings

DEVIEW 2021



Pearson r = 0.77

DEVIEW 2021

A White Box Analysis of ColBERT

ColBERT learns a notion of term importance correlated with IDF

Exact match remains a key component and is promoted for terms with high IDF

We can benefit from IR priors!
Modelling Exact Match is important:
- Design of a sparse retrieval model SPLADE


ECIR 2021

Best Short Paper Award

for the paper titled:

A White Box Analysis of ColBERT

Authors:

Thibault Formal, Stéphane Clinchant and Benjamin Piwowarski

Received at

The 43rd European Conference on Information Retrieval

Online Conference - Lucca | Italy
28th March - 1st April 2021

General Chairs
Raffaele Perego and **Fabrizio Sebastiani**

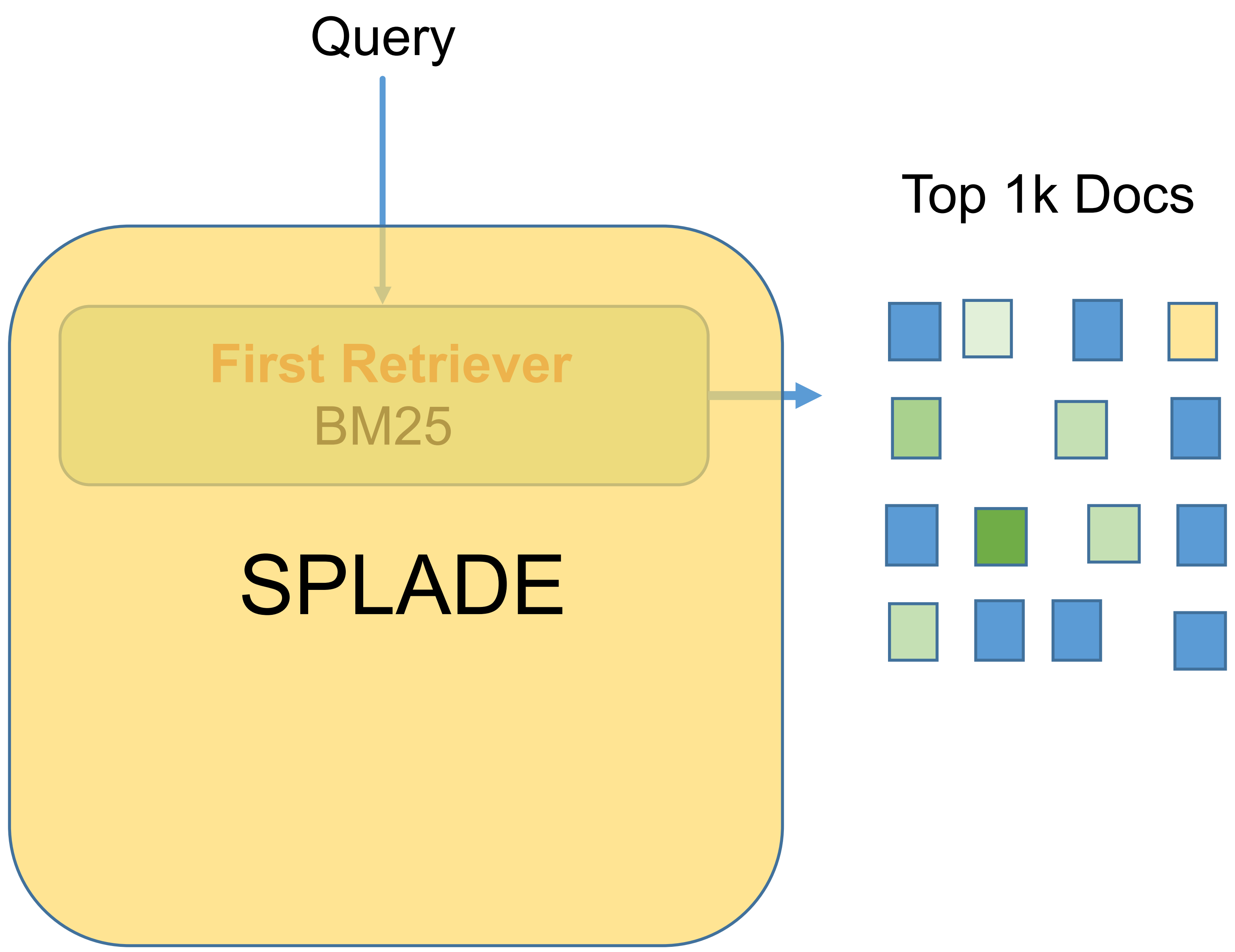
Bloomberg
Engineering

43rd EUROPEAN CONFERENCE ON INFORMATION RETRIEVAL

SPLADE


BM25
sparse
TF-IDF
Vanilla BERT (2019)
RE-RANKING
Siamese BERT (2019)
dense embeddings +
ANN for retrieval
ColBERT (2020)
token-level interactions
ANN for each token
large collection size!
Improved sampling (2020/2021)
• ANCE
• RocketQA
• TAS-balanced
Distillation (2020)
• MarginMSE
• TCT-ColBERT
dense
approaches
...
sparse approaches
DeepCT (2019)
BERT based term re-weighting (regression)
store weights in standard inverted index
doc2query/docT5 (2019)
seq2seq document expansion (predicting q for d)
new collection: index and BM25
Sparse expansion (2020/2021)
• SparTerm
• SPARTA
• SPLADE
predict importance for each term in voc space

First Stage Retriever: SPLADE



Goals:

Infer sparse representations directly

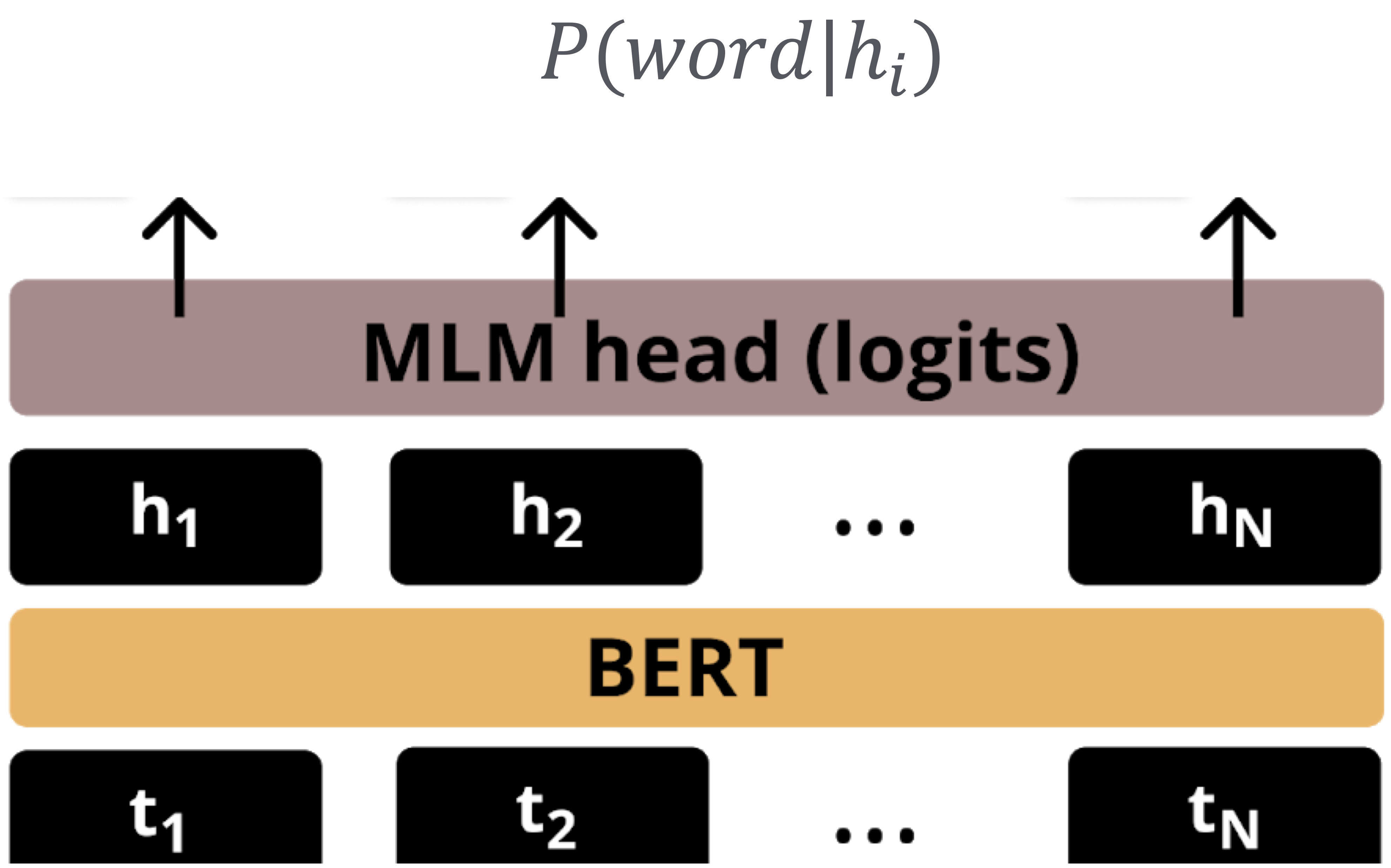
SPLADE:

- Supervised query and document expansion
- Sparse Regularization
- Controllable Sparsity≠ previous approach

SPLADE: BERT and MLM

BERT is already able to perform document expansion naturally

Reuse the MLM head instead of throwing it away!

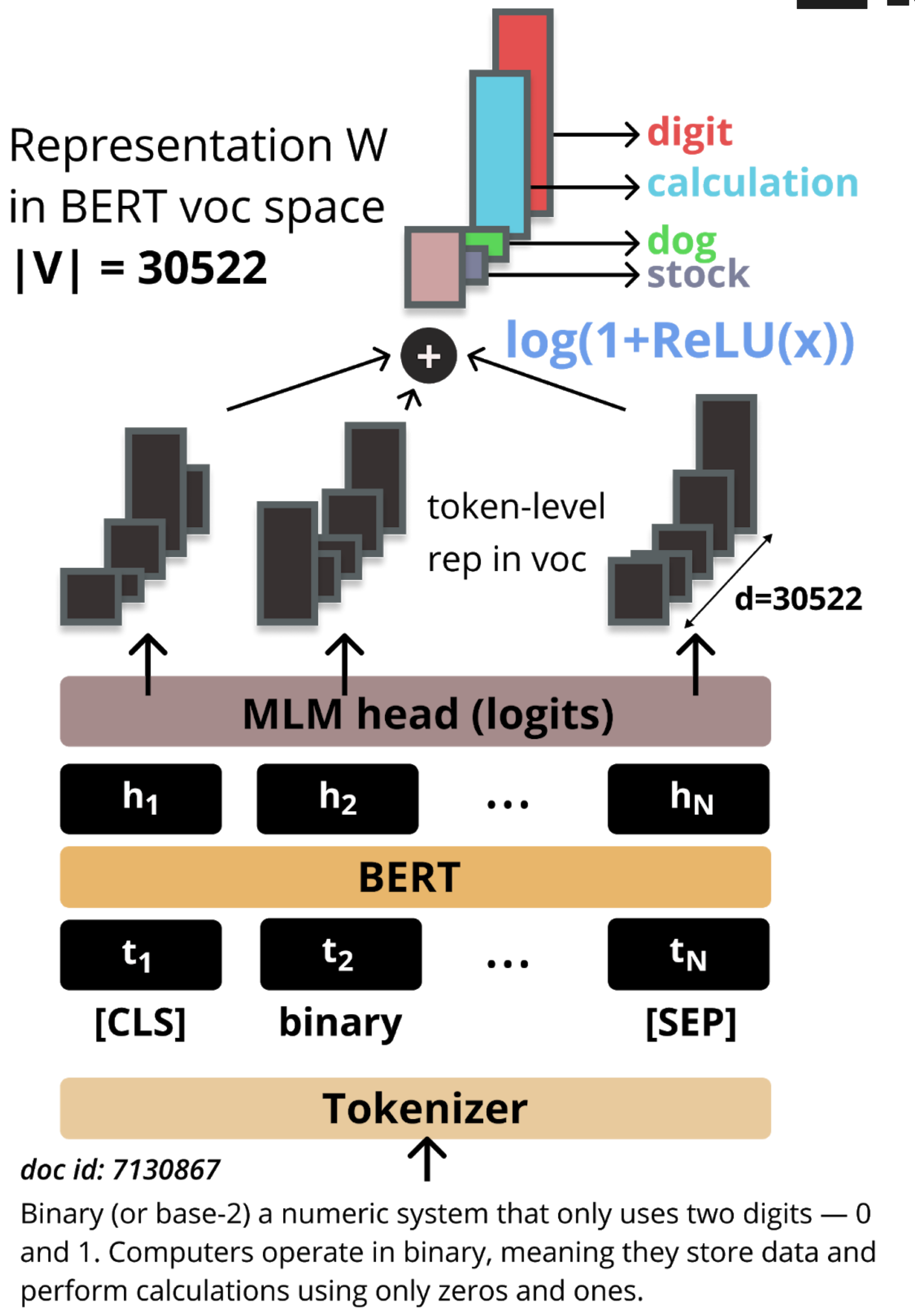


SPLADE: Key Ingredients

No CLS pooling but projecting in BERT vocabulary (with the MLM head)

$$w_{ij} = \text{transform}(h_i)^T E_j + b_j$$

$$w_j = \max_{i \in t} \log\left(1 + \text{ReLU}(w_{ij})\right)$$



SPLADE : Training Loss

Ranking Loss

InfoNCE

$$\mathcal{L}_{rank-IBN} = -\log \frac{e^{s(q_i, d_i^+)}}{e^{s(q_i, d_i^+)} + e^{s(q_i, d_i^-)} + \sum_j e^{s(q_i, d_{i,j}^-)}}$$

SPLADE: Sparse Regularization

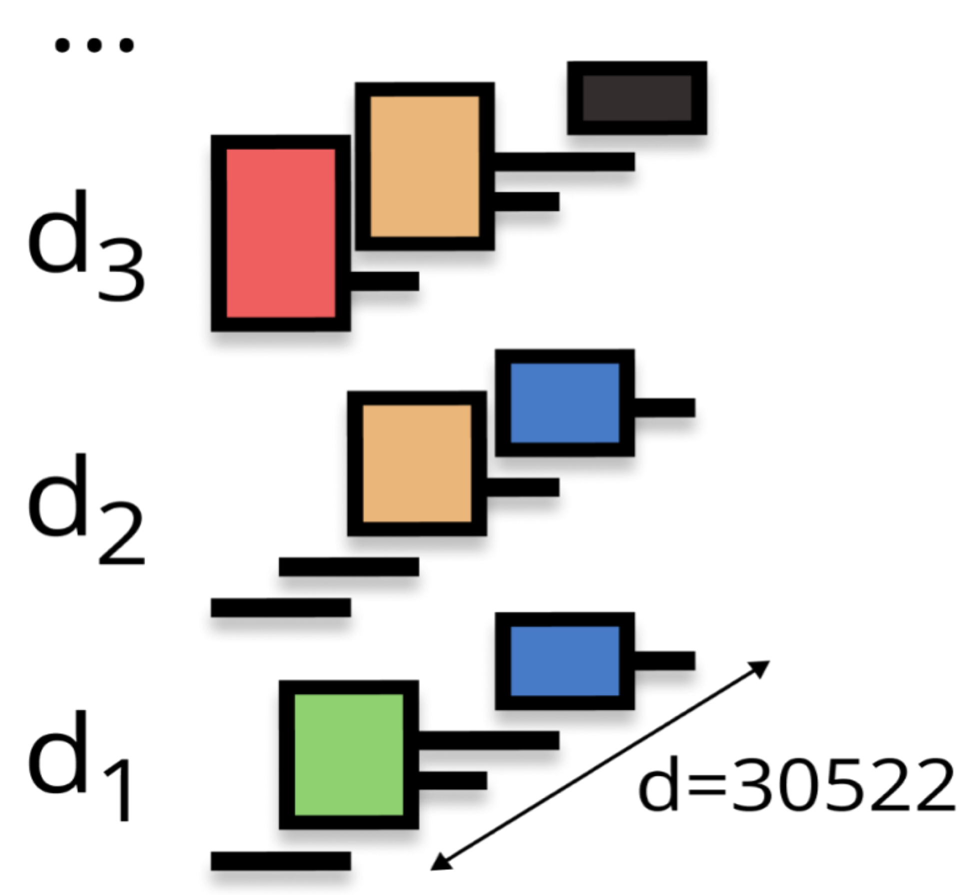
DEVIEW 2021

- Log Activation
- FLOPS Regularization (ICLR'20):

directly optimize a proxy for the number of FLOPS

Main Idea:
'Count the number of activations of a word in a batch'

$$\ell_{\text{FLOPS}} = \sum_{j \in V} \bar{a}_j^2 = \sum_{j \in V} \left(\frac{1}{N} \sum_{i=1}^{N} w_j^{(d_i)} \right)^2$$



SPLADE: Total Loss

Ranking Loss

$$\mathcal{L}_{rank-IBN} = -\log \frac{e^{s(q_i,d_i^+)}}{e^{s(q_i,d_i^+)} + e^{s(q_i,d_i^-)} + \sum_j e^{s(q_i,d_{i,j}^-)}}$$

Sparsity

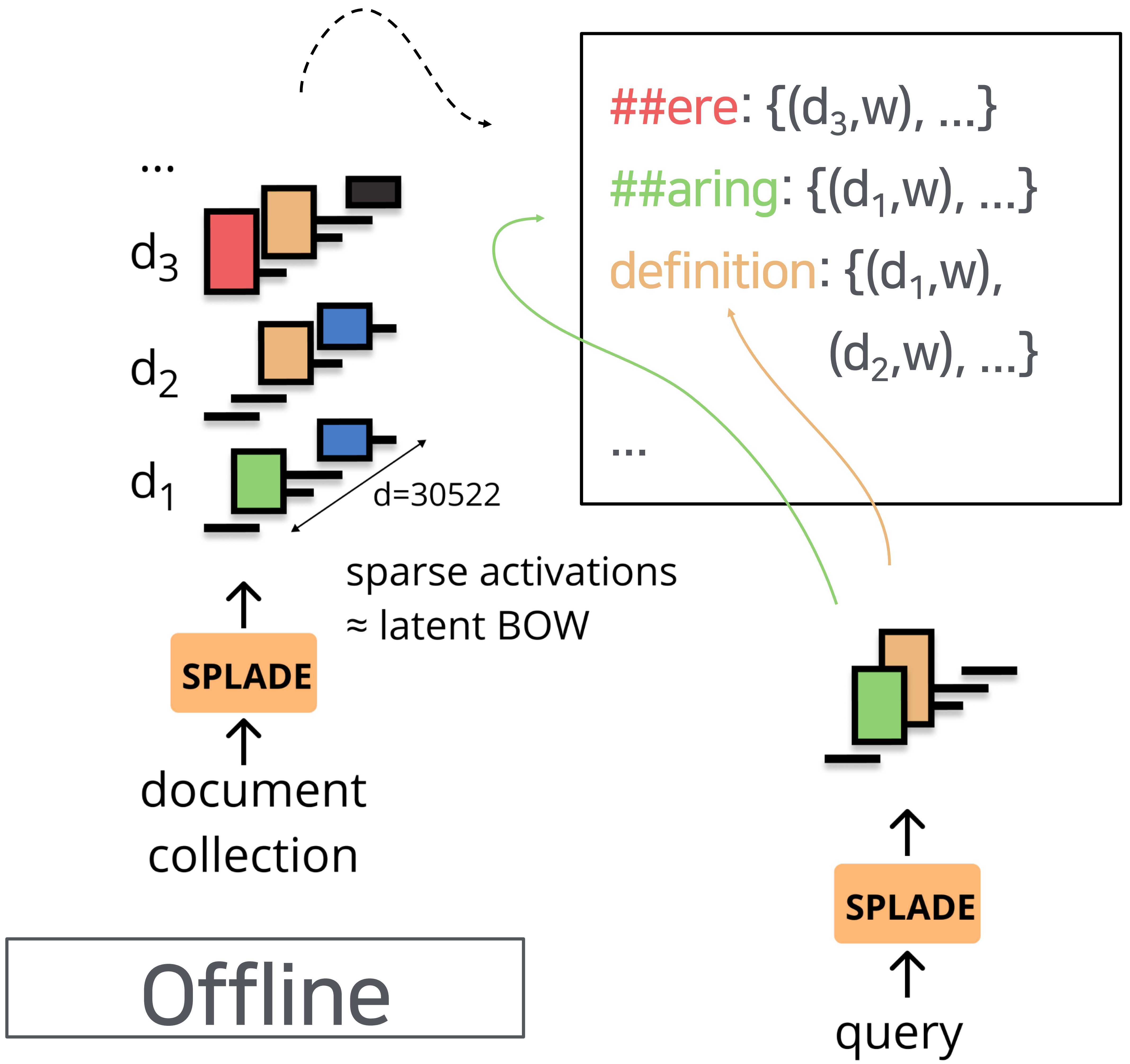
- Log Activation
- FLOPS Regularization (ICLR'20):

directly optimize a proxy for the number of FLOPS

$$\ell_{\text{FLOPS}} = \sum_{j \in V} \bar{a}_j^2 = \sum_{j \in V} \left(\frac{1}{N} \sum_{i=1}^{N} w_j^{(d_i)} \right)^2$$

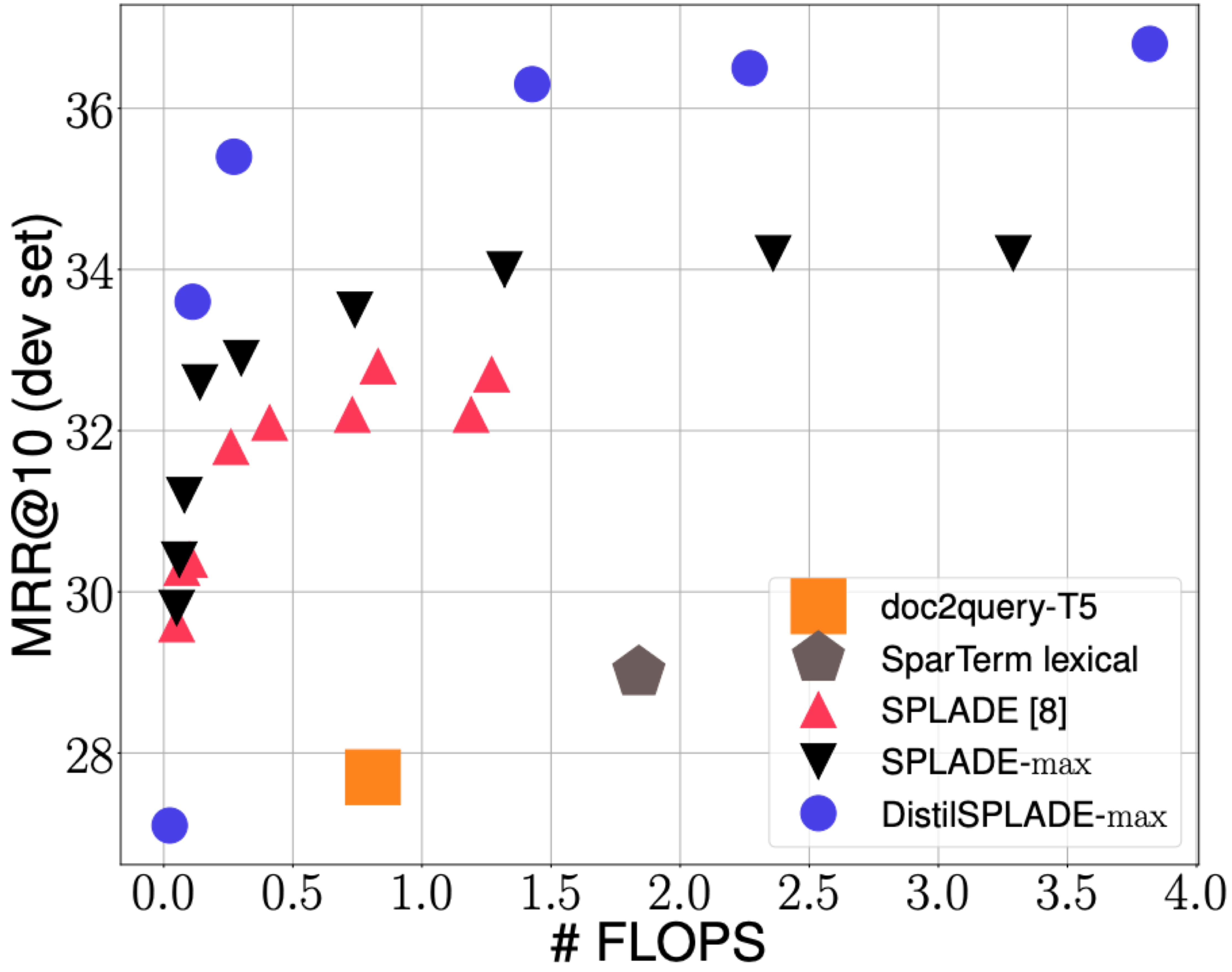
$$\mathcal{L} = \mathcal{L}_{rank-IBN} + \lambda_q \mathcal{L}_{\text{reg}}^q + \lambda_d \mathcal{L}_{\text{reg}}^d$$

Indexing and inference



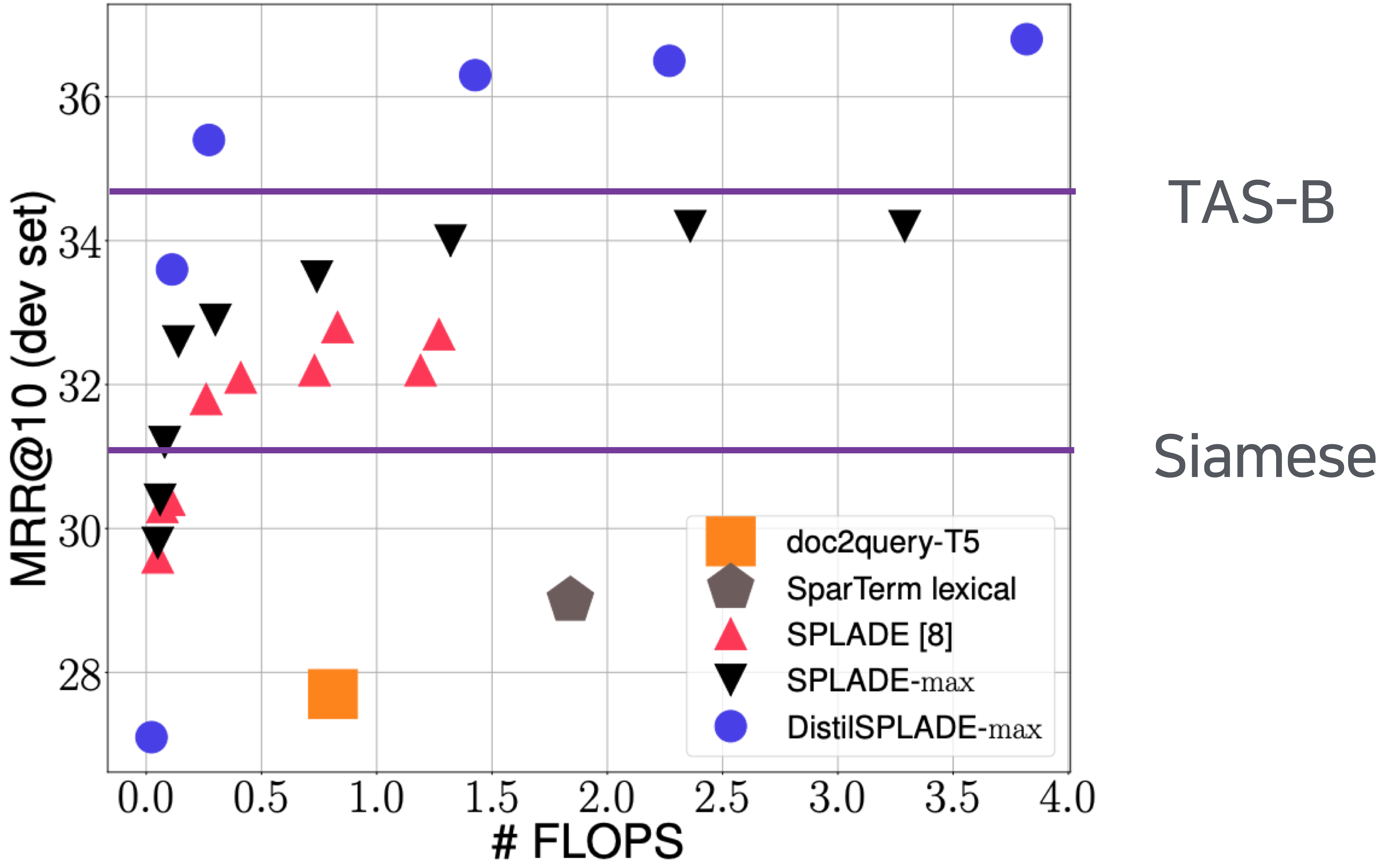
Performance vs FLOPS

Figure 1: Performance vs FLOPS for SPLADE models trained with different regularization strength λ on MS MARCO.



Performance vs FLOPS

Figure 1: Performance vs FLOPS for SPLADE models trained with different regularization strength λ on MS MARCO.



SPLADE Experiments: MS-Marco and TRECDL'19

Model	MRR@10 MSMARCO Dev	NDCG@10 TREC DL19
BM25	19.4	50.1
docT5	27.7	64.2
Siamese Bert	31.2	63.7
TAS-B	34.7	71.7
Distill-SPLADE	36.8	72.9

The first Sparse Model that rivals Dense Siamese BERT Models

An example

original document (doc ID: 7131647)

if (1.2) bow (2.56) legs (1.18) ~~is~~ caused (1.29) by (0.47) ~~the~~ bone (1.2) alignment (1.88) issue (0.87) ~~than you may be~~ able (0.29) ~~to~~ correct (1.37) through (0.43) *bow legs* correction (1.05) ~~exercises~~. ~~read more here.~~ *if bow legs is caused by the bone alignment issue than you may be able to correct through bow legs correction exercises.*

stemming effect

bad expansion terms ! **expansion terms** good expansion terms

(leg, 1.62) (arrow, 0.7) (exercise, 0.64) (bones, 0.63) (problem, 0.41) (treatment, 0.35) (happen, 0.29) (create, 0.22) (can, 0.14) (worse, 0.14) (effect, 0.08) (teeth, 0.06) (remove, 0.03)

BEIR Conclusion

BM25	Colbert	TAS-B
45.3	45.6	43.7

- Rerankers transfer well
- Colbert ok too
- Standard siamese don't

"Our results show BM25 is a robust baseline … In contrast, Dense-retrieval models […] often underperform other approaches, highlighting the considerable room for improvement in their generalization capabilities "

SPLADE on BEIR (Zero Shot Benchmark)

DEVIEW 2021

Does SPLADE generalize well to other collections?

BEIR Benchmark: NDCG@10 for available collections

TAS-B : SOTA (August'21) Dense Bi-Encoder Retrieval Model

BM25	Colbert	TAS-B	SPLADE	Distill-Splade
45.3	45.6	43.7	46.4	50.6

Conclusion

CONTENTS

Sparse Lexical AnD Expansion Model for First Stage Retrieval

The first Sparse Model that rival Dense ones

Summary

SPLADE, an efficient, FLOPS- controllable, interpretable, first stage retriever, that transfers well

https://github.com/naver/splade

Future work?

Join us!

Multiple positions in the Search and Recommendation team at NAVER LABS Europe

https://europe.naverlabs.com/careers/

NAVER LABS Europe, Grenoble, France

Q & A

Thank You