Constructing set-compositional and negated representations for first-stage ranking

Antonios Minas Krasakis

Andrew Yates

Evangelos Kanoulas

University of Amsterdam

Introduction

- Large ranking datasets (e.g. MSMarco) typically comprise of **simple ad-hoc queries.**

- Focus on complex **set-compositional and negated queries**:
  - e.g. *"Books about French monarchs but not Napoleon"*
  - enable users to:
    - expressing more complex information needs
    - discovery of niche items

Challenges

1

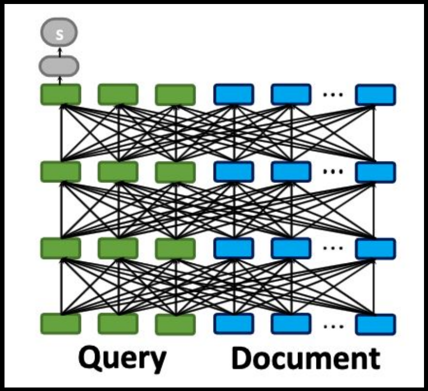
Negations, conjunctions, disjunctions:

Traditionally hard to handle in NLP and particularly IR

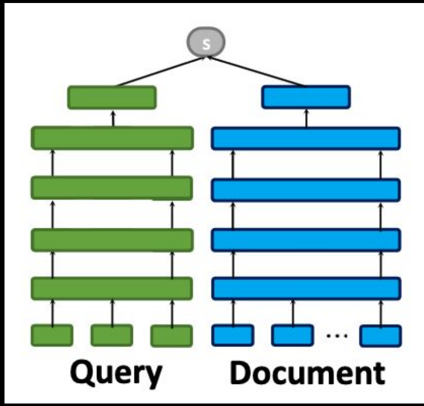
2

First stage ranking

No cross-attentions between query and document; Compositionality, negation needs to be encapsulated in representation



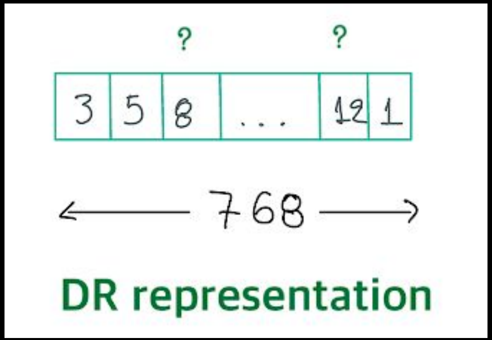
Reranker



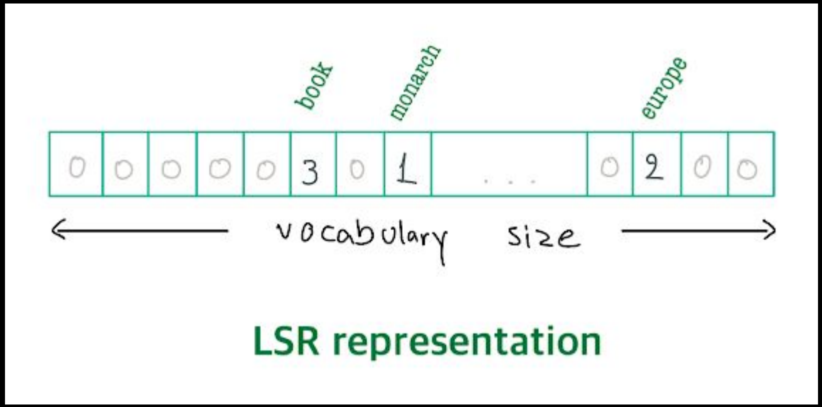
First-stage ranker

Learned Sparse Retrieval (LSR) models: a primer

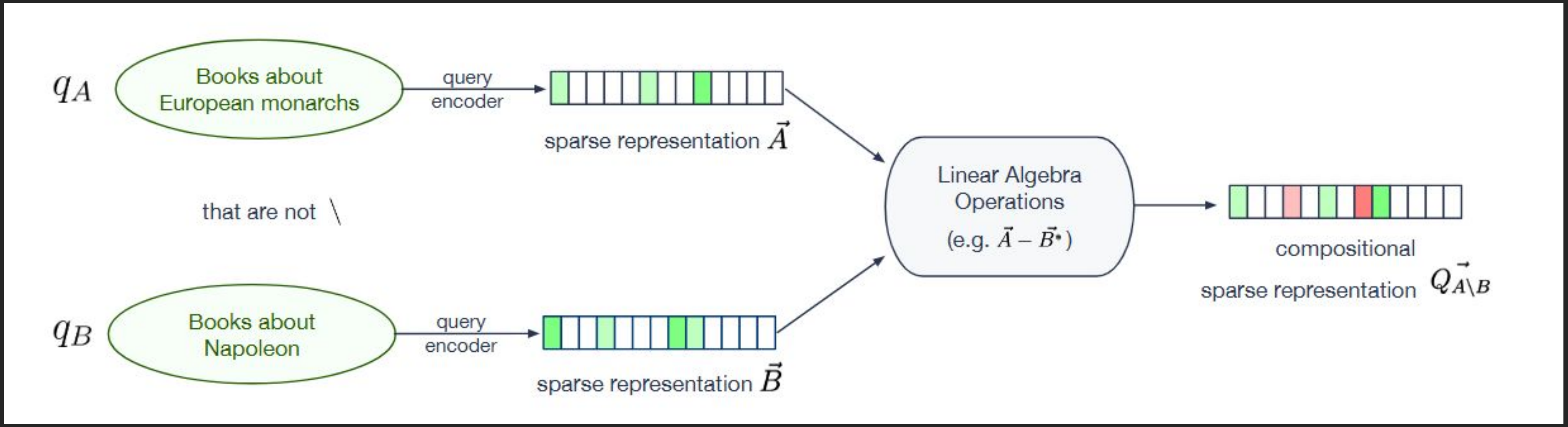
- are SOTA retrieval models
    - outperform DR in out-of-domain settings



- with lexically grounded representations
    - each dimension is linked to a (BERT) token



RQ1: Can we construct zero-shot compositional representations?



Introduce **zero-shot framework** that creates lexically grounded **Learned Sparse Retrieval (LSR)** representations, using **intuitive Linear Algebra Operations** (LAO)

Linear Algebra Operations (LAO) for zero-shot compositional query representations

Disentangled Negation

(for set difference)

$$f_{CQE}(A \setminus B) = \vec{A} - \vec{B}^*$$

$$\text{,where } \vec{B}^* = \vec{B} - \mathbb{1}_A \odot \vec{B}$$

Penalizes negative query terms not appearing on the positive query, e.g.

{ books:, + 2.1,
French: + 2.9,
Napoleon: - 3.1, ...}

Combined Pseudo Terms

(for intersections)

$$f_{CQE}(A \cap B) = \vec{A} \otimes \vec{B}$$

Constructs synthetic bi-gram tokens, e.g.:

{
(Books∩Napoleon):
+ 3.2,
...}

RQ1: zero-shot compositional representations

- Set difference:
  - Disentangled negation consistently outperforms or on par with other methods
- Intersections:
  - CPT outperforms all methods
  - still challenging

method	$Q_{\vec{A \star B}}$	nDCG@10	R@100
Set Difference ($q_A \setminus q_B$)			
vanilla	$f_{QE}(q_A$ but not $q_B)$	0.132	0.307
Subtraction	$\vec{A} - \vec{B}$	0.120	0.190
Ignore negation	$\vec{A}$	0.233	**0.390**
Orthogonal negation [48]	$\vec{A} - \pi_{\vec{B}}\vec{A}$	0.235	0.379
NRF [44]	$\vec{A} - \lambda \cdot \vec{B}$	**0.256**	0.379
Disentangled negation **(ours)**	$\vec{A} - \vec{B}^*$	**0.258**	**0.398**
Intersection ($q_A \cap q_B$)			
vanilla	$f_{QE}(q_A$ that are also $q_B)$	0.063	0.209
Addition	$\vec{A} + \vec{B}$	0.068	0.222
Max Pool	MaxPool$(\vec{A}, \vec{B})$	0.069	0.238
CPT **(ours)**	$\vec{A} \otimes \vec{B}$	**0.080**	**0.268**

RQ2: What happens if we train retrievers using compositional queries?

- after training, LSR & DR improve for negations, but not intersections

train_set	Set Difference		Intersection	
	nDCG@3	*R@100*	*nDCG@3*	*R@100*
	Learned Sparse Retrieval			
no compositional queries [a]	0.132	0.307	0.063	0.209
with compositional queries [b]	0.240^{a}	$\mathbf{0.409}^{a}$	0.062	0.227
zero-shot, with LAO [c] (best of RQ1)	$\mathbf{0.258}^{a}$	0.398^{a}	$\mathbf{0.080}^{a,b}$	$\mathbf{0.268}^{a,b}$
	Dense Retrieval			
no compositional queries [d]	0.154	0.356	**0.088**	**0.290**
with compositional queries [e]	$\mathbf{0.251}^{d}$	$\mathbf{0.450}^{d}$	0.081	0.283

RQ2: What happens if we train retrievers using compositional queries?

- after training, LSR & DR improve for negations, but not intersections
- zero-shot methods competitive or better than trained models

train_set	Set Difference		Intersection	
	nDCG@3	*R@100*	*nDCG@3*	*R@100*
	Learned Sparse Retrieval			
no compositional queries [a]	0.132	0.307	0.063	0.209
with compositional queries [b]	0.240^{a}	$\mathbf{0.409}^{a}$	0.062	0.227
zero-shot, with LAO [c] (best of RQ1)	$\mathbf{0.258}^{a}$	0.398^{a}	$\mathbf{0.080}^{a,b}$	$\mathbf{0.268}^{a,b}$
	Dense Retrieval			
no compositional queries [d]	0.154	0.356	**0.088**	**0.290**
with compositional queries [e]	$\mathbf{0.251}^{d}$	$\mathbf{0.450}^{d}$	0.081	0.283

How do LSR representations look like?

Set-difference / Negations

- Trained models learn to ignore negated terms
- Disentangled Negation **penalizes** negated terms

method	representation term scores
	Books about non-European monarchs
vanilla	(monarch##: 1.11, **not: 1.1**, book##: 1.03, **european: 1.02**, king: 0.38, non: 0.35, ...)
Disentangled Negation	(monarch##: 1.25, history: 1.13, book##: 1.02, king: 0.77, ..., **europe: -0.66**, **european: -1.17**
finetuned (RQ2)	(monarch##: 1.2, king##: 1.04, books: 0.78, tsar: 0.75, encyclopedia: 0.72, book: 0.71, biography: 0.67, emperor: 0.65, ..., **~~european: 0.0~~)**
	(Documentary films about education) ∩ *(Documentary films about intellectual disability)*
vanilla	(intellectual: 1.14, disability: 1.07, documentary: 1.04, education: 1.04, film##: 0.9,disabled: 0.82, educational: 0.73, ...)
CPT	(**education** ∩ **intellectual**: 1.57, documentary ∩ intellectual: 1.49, **education** ∩ **disability**, 1.47, documentary ∩ disability: 1.40, ...)
finetuned (RQ2)	documentary: 1.21, education: 1.12, disability: 1.08, intellectual: 1.02, disabled: 1.02, school: 0.77, academic: 0.73, autism: 0.67, ...

How do LSR representations look like?

Intersections

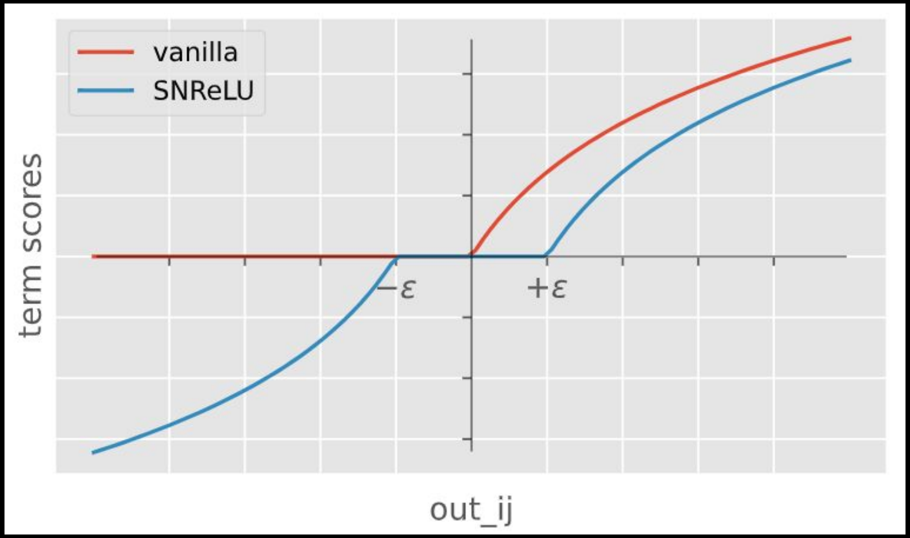
- CPT: the only effective in capturing term co-occurrence (e.g. education ∩ disability)

method	representation term scores
	Books about non-European monarchs
vanilla	(monarch##: 1.11, **not: 1.1**, book##: 1.03, **european: 1.02**, king: 0.38, non: 0.35, ...)
Disentangled Negation	(monarch##: 1.25, history: 1.13, book##: 1.02, king: 0.77, ..., **europe: -0.66**, **european: -1.17**
finetuned (RQ2)	(monarch##: 1.2, king##: 1.04, books: 0.78, tsar: 0.75, encyclopedia: 0.72, book: 0.71, biography: 0.67, emperor: 0.65, ..., **~~european: 0.0~~)**
	(Documentary films about education) ∩ (Documentary films about intellectual disability)
vanilla	(intellectual: 1.14, disability: 1.07, documentary: 1.04, education: 1.04, film##: 0.9,disabled: 0.82, educational: 0.73, ...)
CPT	(**education** ∩ **intellectual**: 1.57, documentary ∩ intellectual: 1.49, **education** ∩ **disability**, 1.47, documentary ∩ disability: 1.40, ...)
finetuned (RQ2)	documentary: 1.21, education: 1.12, disability: 1.08, intellectual: 1.02, disabled: 1.02, school: 0.77, academic: 0.73, autism: 0.67, ...

RQ3: Can we improve LSR for negations?

Re-design the activation function of LSR models (splade) to allow for **negative** term weights, while preserving:

- log-saturation effect
- sparsity



RQ3: Can we improve LSR for negations?

NevIR dataset

- vanilla performance: worse than random
- improvements with negative weights: ~80%

Model	FT on NevIR	Negative term scores	+/- weight aggreg.	Pairwise Accuracy
random	✗	–	–	25.00%
splade	✗	✗	–	7.88%
	✓	✗	–	23.07%
	✓	✓	$\pm\|max\|$	37.09%
	✓	✓	*sum*	**42.73%**

Table 5: Performance of LSR models trained with negative term scores on NevIR.

RQ3: Can we improve LSR for negations?

NevIR dataset

- vanilla performance: worse than random
- improvements with negative weights: ~80%

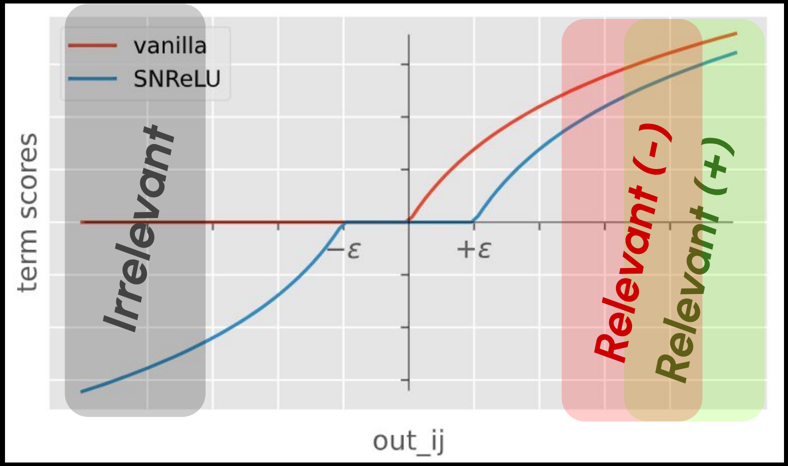
QUEST dataset

- improves NDCG in negated queries
- performance drop for non-negated

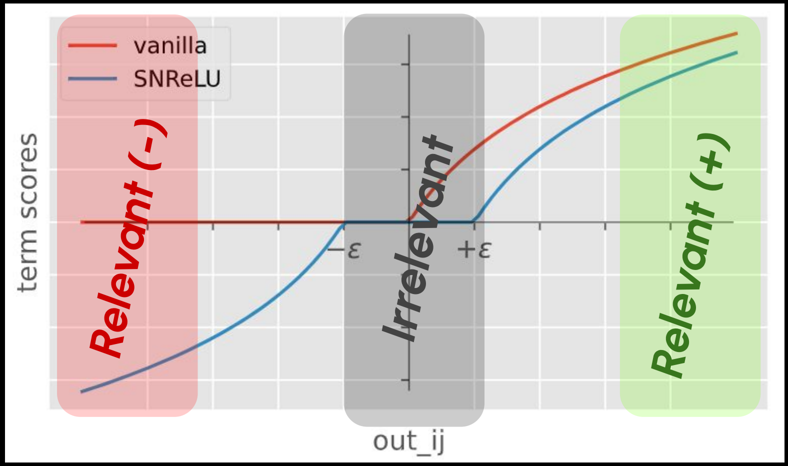
Negative scores	+/- weight aggreg.	Atomic nDCG@10	Atomic R@100	Set Difference nDCG@10	Set Difference R@100
✗	–	**0.284**	**0.540**	0.205	**0.413**
✓	$\pm\lvert max\rvert$	0.182	0.418	0.210	0.340
✓	sum	0.165	0.391	**0.218**	0.351

Table 6: Performance of LSR models trained with negative term scores on Quest.

Intuition: Why is it hard to adapt LSR models for negations?



pre-training
(BERT, MSMarco)



architecture with
negative term scores

Conclusions

Introduced a zero-shot framework for modelling negations & intersections

vanilla LSR & DR models (partly) learn modeling negations, but fail on intersections

Negative term weights can improve LSR representations

paper link