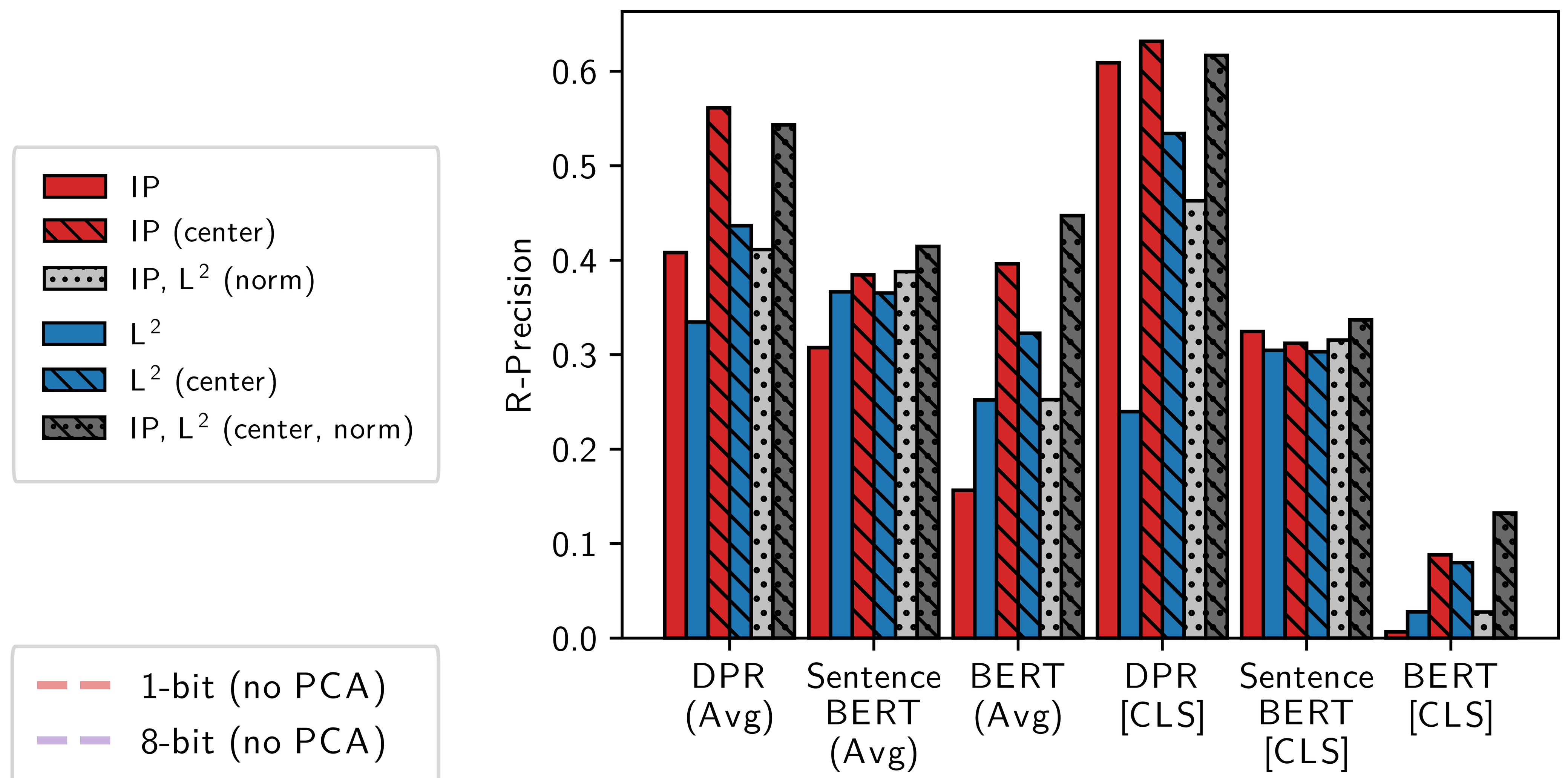
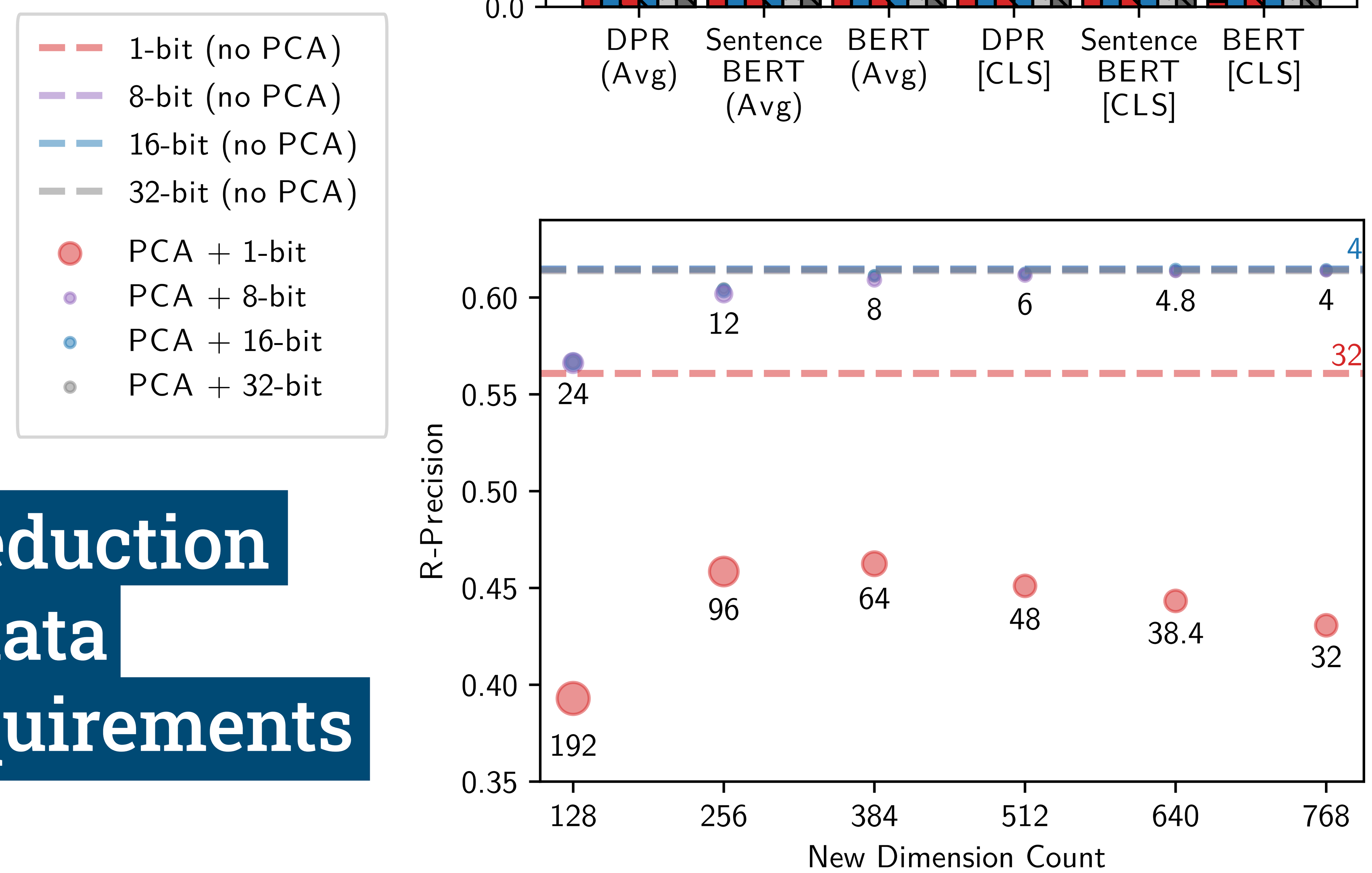
Knowledge Base Index Compression via Dimensionality and Precision Reduction

Vilém Zouhar, Marius Mosbach,
Miaoran Zhang, Dietrich Klakow
{Saarland University}
{Spa-NLP @ ACL 2022}

1.	$Z = \arg\,\text{top-k}_{d \in D}\ \text{rel.}(q, d)$	Retrieve top k documents
2.	$\text{rel.}(q, d) \approx \text{sim}(f_Q(q), f_D(d))$	Approximate relevancy by vector similarity on embeddings from Bert/SentenceBert/DPR
3.	$\approx \text{sim}(r_Q(f_Q(q)), r_D(f_D(d)))$	Reduce embedding dimensionality using functions r_Q, r_D (queries and docs)

Method	Compression	Original		Center +Norm.
		IP	L^2	{IP, L^2} (% original)
Original	1×	0.609	0.240	0.618 (100%)
Sparse Projection (128)	6×	0.398	0.448	0.457 (74%)
Dimension Dropping (128)	6×	0.426	0.466	0.478 (77%)
Greedy Dimension Dropping (128)	6×	0.447	0.478	0.504 (82%)
PCA (128)	6×	0.577	0.562	0.579 (94%)
PCA (128, scaled top 5)	6×	0.586	0.572	0.592 (96%)
Autoencoder (128, single layer)	6×	0.585	0.569	0.588 (95%)
Autoencoder (128, shallow decoder)	6×	0.599	0.582	0.599 (97%)
Autoencoder (128, single layer) $+L_1$	6×	0.600	0.587	0.601 (97%)
Autoencoder (128, shallow decoder) $+L_1$	6×	0.601	0.591	0.601 (97%)
Precision 16-bit	2×	0.612	0.610	0.615 (100%)
Precision 8-bit	4×	0.613	0.610	0.614 (99%)
Precision 1-bit (offset 0 5)	32×	0 559	0 556	0 561 (91%)
PCA (245) +Precision 1-bit (offset 0.5)	100×	0.459	0.458	0.461 (75%)
PCA (128) +Precision 8-bit	24×	0.558	0.553	0.567 (92%)





0. Always center & normalize before & after dimension reduction
1. PCA: quick and good-enough solution requiring little data
2. Autoencoder: slightly better, less stable, larger data requirements
3. 8/16-bit precision: almost no performance loss