ETH zürich

ETH AI CENTER

THE ITALIAN JOB




ID 04

WE Initialization Improves KG Link Prediction

Piersilvio De Bartolomeis[1], Antonio Lopardo[2], Luca Malagutti[3], Alessandro Ruzzi[4], Carl Allen
Department of Computer Science, ETH Zurich

1 Introduction

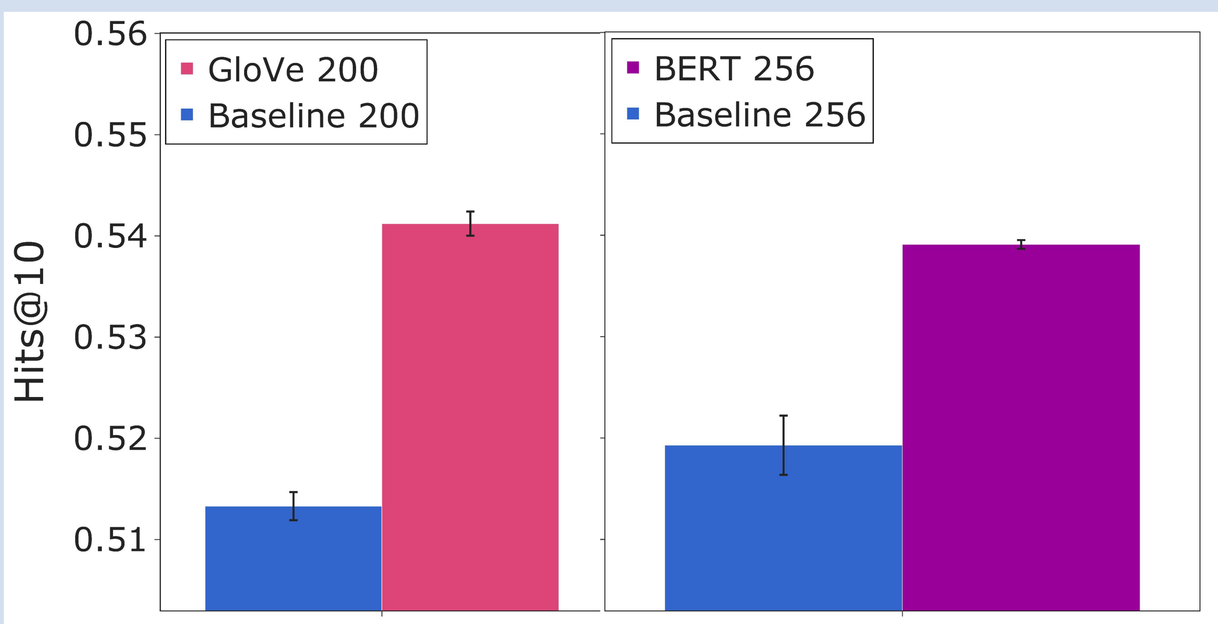
Link Prediction entails predicting relations between entities of a Knowledge Graph (KG).

Word Embeddings (WE) are low dimensional semantic word representations learned from a large text corpora. We experimented with Word2vec, GloVe (co-occurrence based) and BERT-mini (Transformer-based).

The Project aim is to improve KG link prediction by initialising KG models' entity embeddings with WE.

2 Main Contribution[1234]

Initializing KG's entities with word vectors significantly improves link prediction performance.



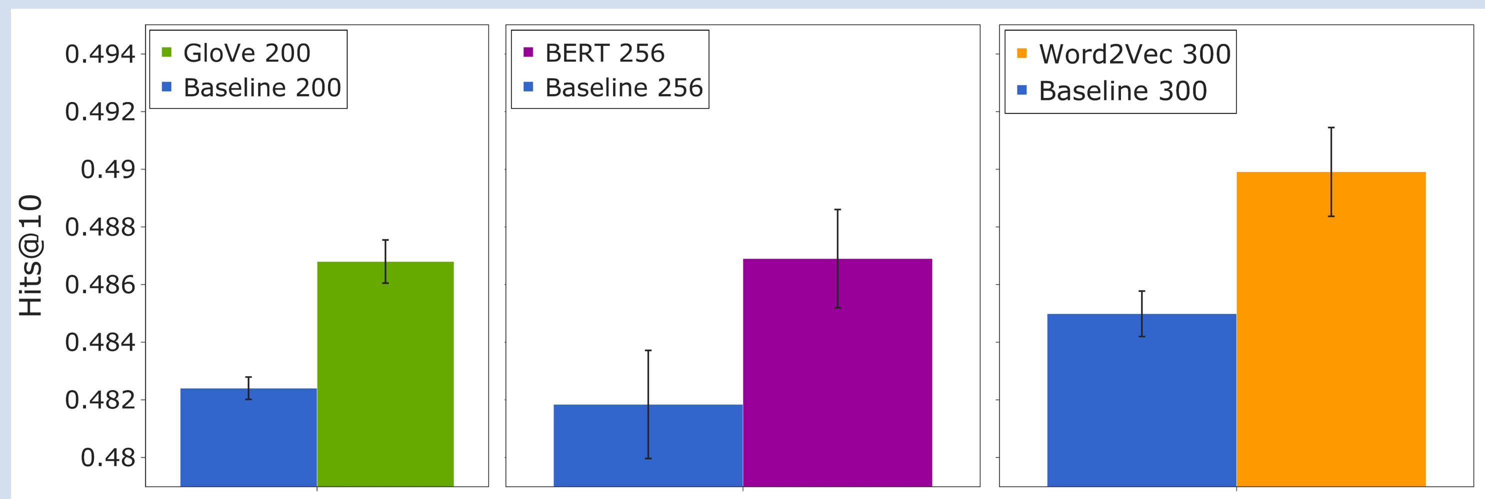
(Left) We compare TuckER initialized at random (baseline) vs. TuckER initialized with GloVe embeddings.
(Right) We compare TuckER initialized at random (baseline) vs. TuckER initialized with BERT embeddings.

TuckER is a KG model that decomposes the implicit "adjacency tensor" of the KG, in a manner similar to SVD.
WordNet18-RR is a link prediction dataset composed of 93k triples/"facts", 41k entities, and 11 relations.
Experimental Setup: For each initialization, we tune all the model hyper-parameters, (e.g. dropouts) and run each model 5 times with early stopping.

3 Generalization over Models[34]

We reproduce our experiments on *WN18-RR* with **DistMult**, a KG model that forces all relation embeddings to be diagonal matrices.

WE entity initializations benefit a variety of models.

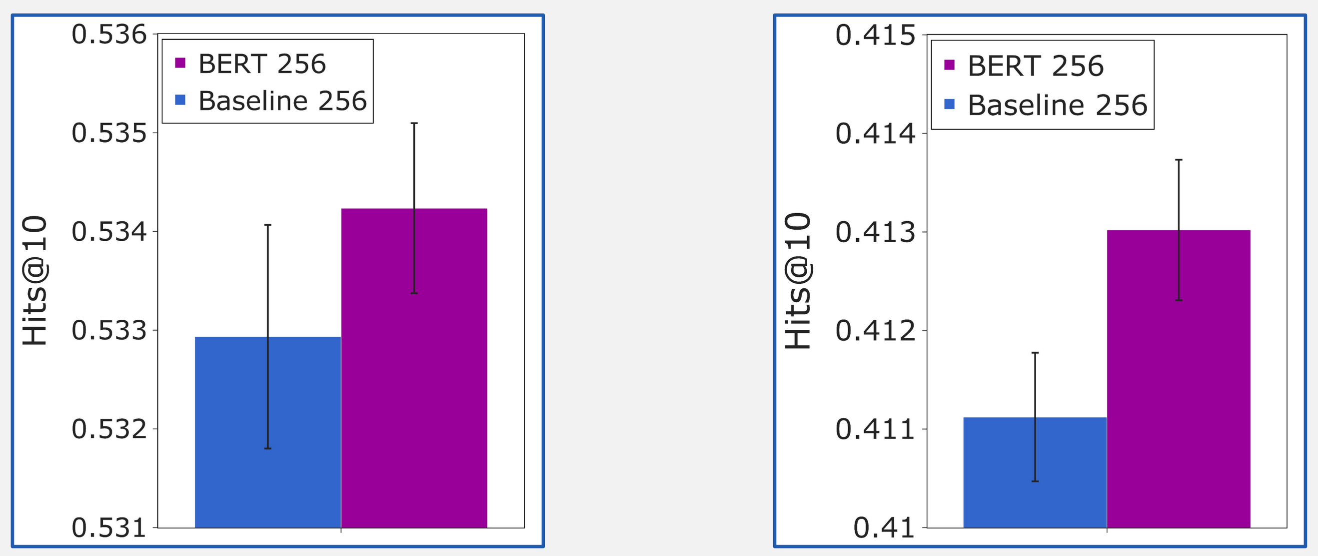


4 Generalization over Datasets[14]

We perform similar verification experiments with the **FreeBase15k237** dataset, made of fewer entities (14,505) but significantly more relations (237).

On multiple models and datasets, WE initialization improves link prediction performance.

On *FB15k-237* the models improve less than on *WN18-RR* due to the increased semantic complexity of the entities, as they include places and people's names.



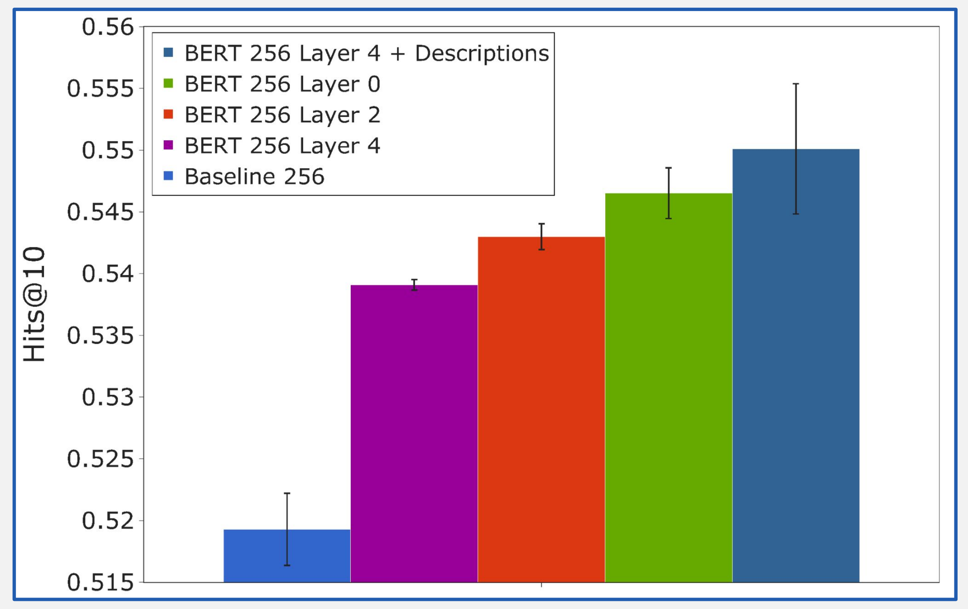
TuckER **DistMult**

5 Enhancing BERT Initialization[23]

We initialize entity vectors of TuckER with embeddings taken from BERT at different depths, observing that **lower-level representations perform better.**

We leverage BERT contextual and token embeddings:

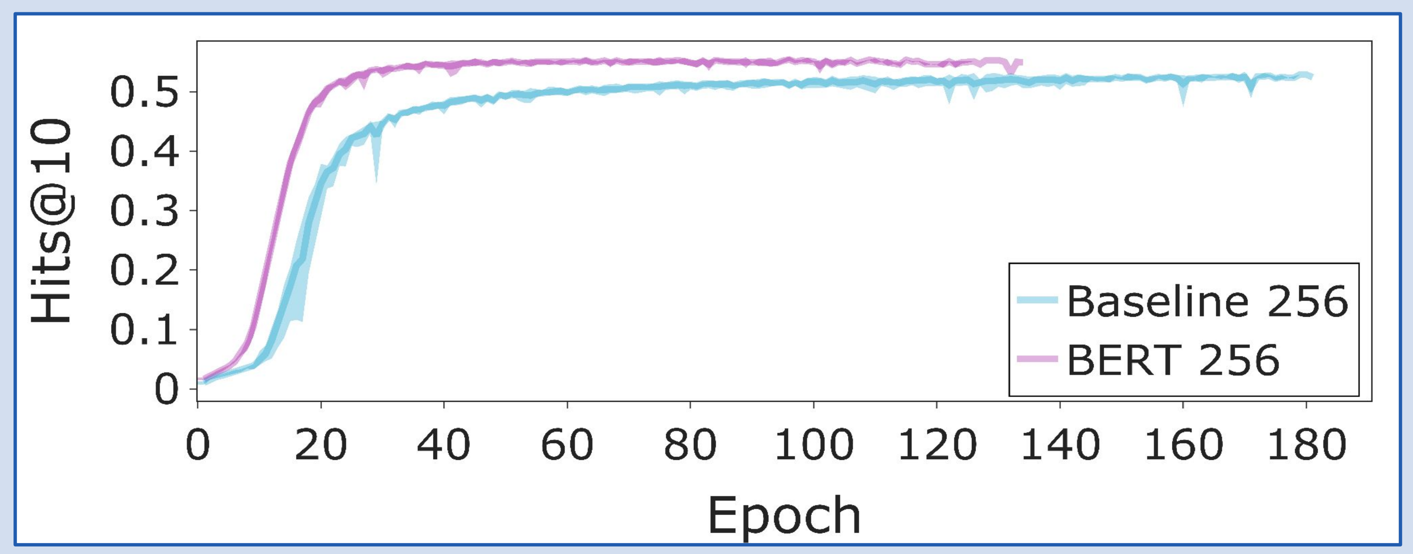
- Adding an entity definition to BERT's input improves performance, especially at deeper layers.
- Giving more weight to a word stem when combining tokens yields non-significant improvements.



6 Convergence Rate[12]

We plot mean and standard deviation of Hits@10 during training. We find that:

Convergence speed significantly increases when entity embeddings are initialized with pre-trained WE



(Blue) TuckER with embedding dimension 256 for both entities and relations, initialized at random.
(Purple) TuckER with embedding dimension 256 for both entities and relations, initialized with BERT.