- **Part 1: Knowledge Graphs and Entities**
  - Welcome & Motivation (Dietz)
  - Knowledge Graphs and GPT (Bast)
  - Entity Linking (Bast)
- **Part 2: Neuro-Symbolic Foundations**
  - Ranking Wikipedia Entities / Aspects (Chatterjee) ← We are here
  - Neural Text Representations and Semantic Annotations (Dietz)
  - Infusion of Symbolic Knowledge into Text Representation (Nie)
- **Part 3: Reasoning, Robustness, and Relevance**
  - Denoising Dense Representations with Symbols (Nogueira)
  - Reasoning about Relevance (Dalton)
  - From PRF to Retrieval Enhanced Generation (Dietz)
- **Part 4: Emerging Topics**
  - Conclusion and Outlook
  - Panel Discussion

Material: https://github.com/laura-dietz/neurosymbolic-representations-for-IR/SIGIR23/

Entity Representations and Entity Ranking

DR. SHUBHAM CHATTERJEE

RESEARCH ASSOCIATE

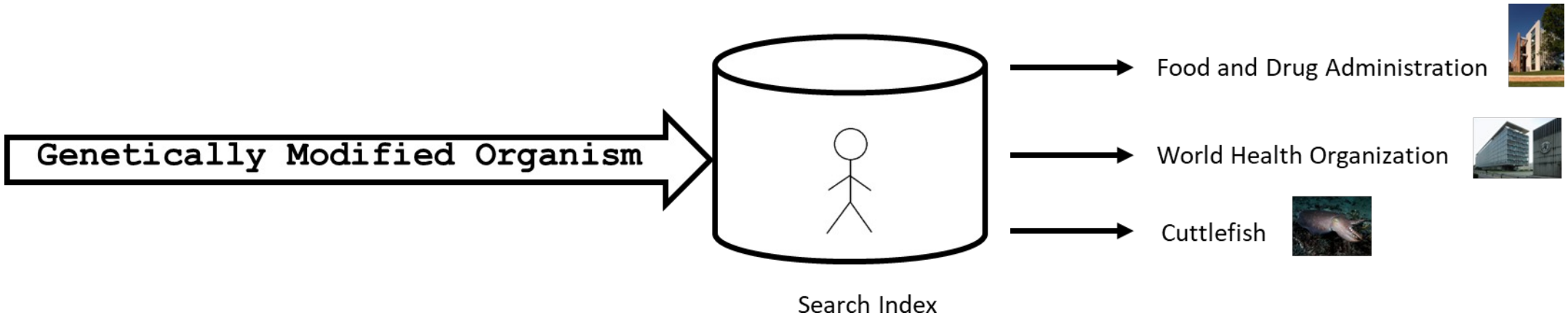
DEPARTMENT OF COMPUTING

SCIENCE UNIVERSITY OF GLASGOW,

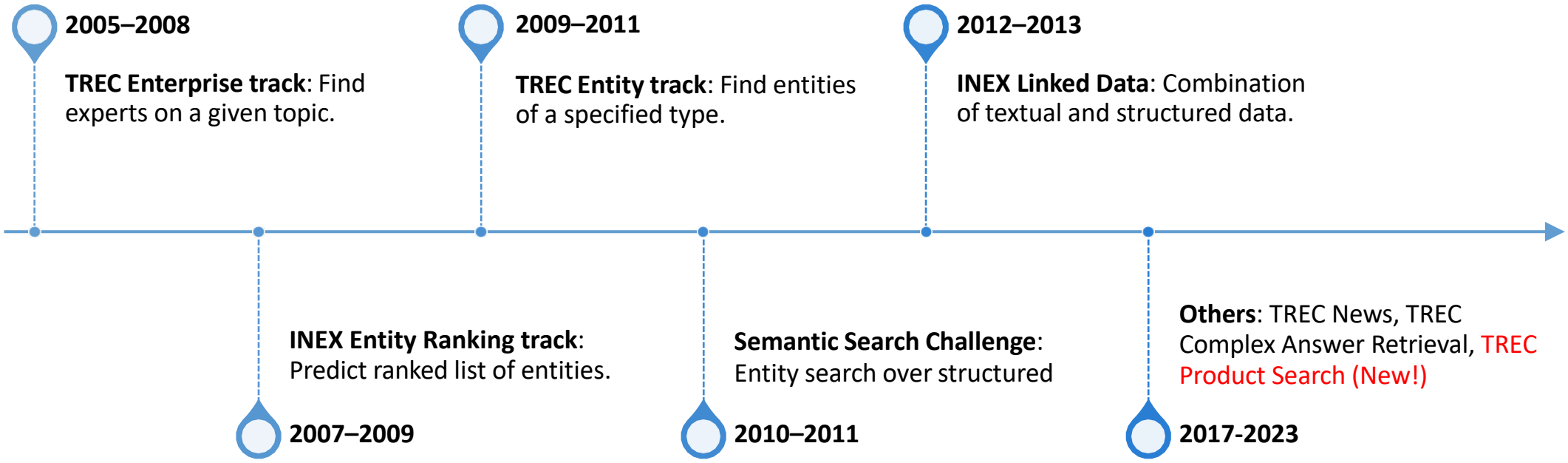
UK

Task: Entity Ranking

Given a query and a KG, retrieve entities that are relevant to the query ordered by the relevance of each entity to the query.



Entity Ranking: Over the Years

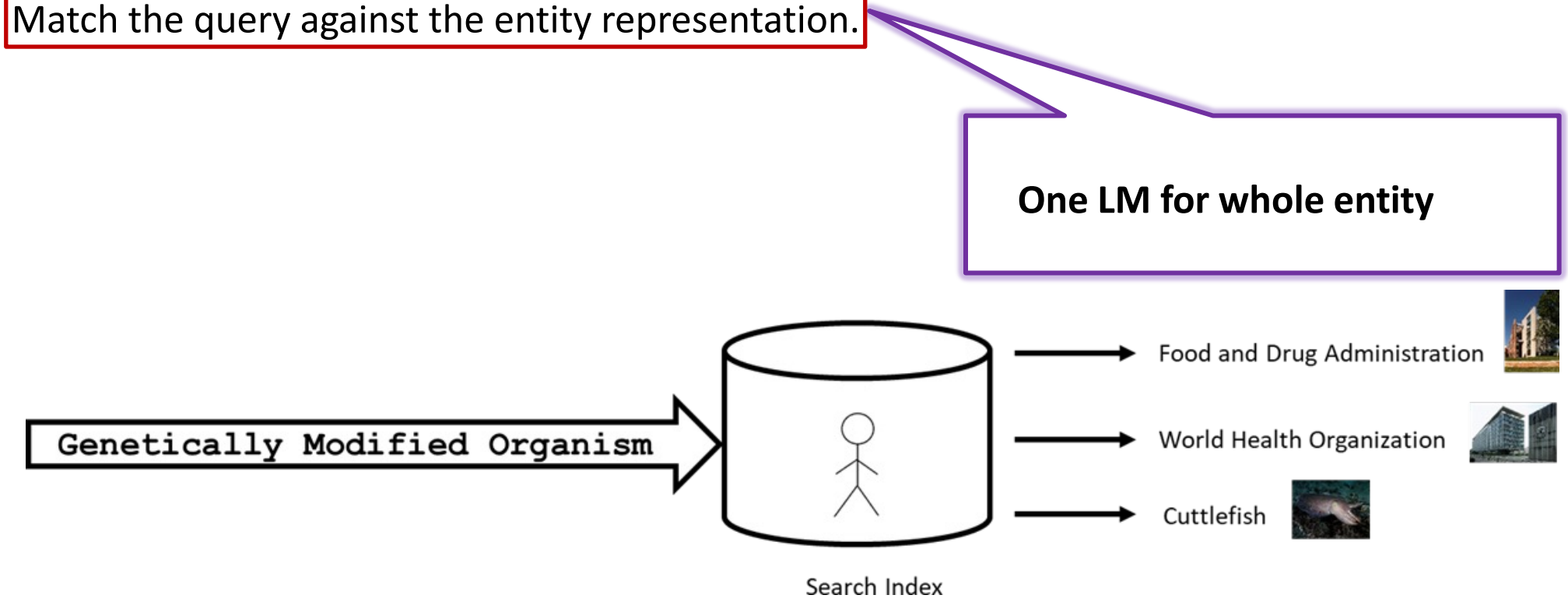


How Do Systems Usually Retrieve Entities?

Probabilistic Unstructured Models

Consider an entity as a single "document".

- ❑ Create a search index of all entities.
- ❑ Match the query against the entity representation.



Probabilistic Unstructured Models

Consider an entity as a single "document".

- ❑ Create a search index of all entities.
- ❑ Match the query against the entity representation.

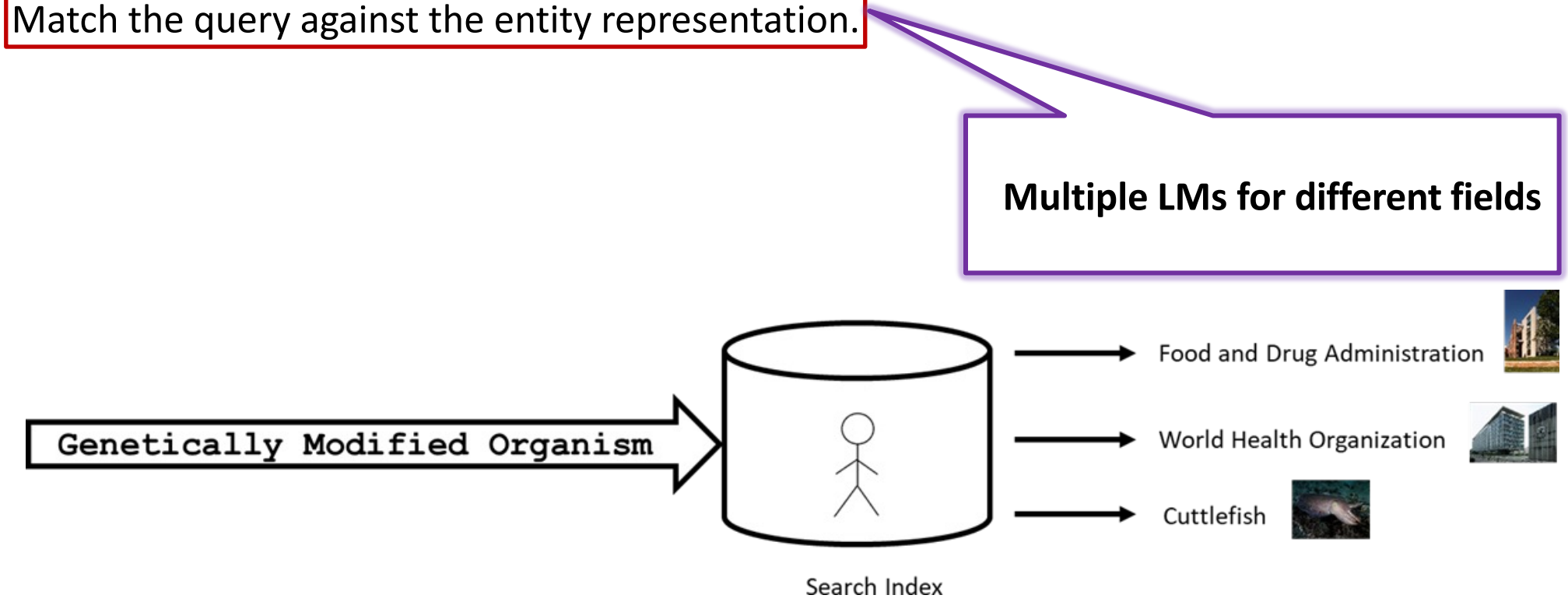
One LM for whole entity

- ❑ **Probability distribution over sequences of words.**
- ❑ **Generate probabilities by training on text corpora in one or many languages.**

Fielded Models

Consider an entity as a "document" with multiple fields.

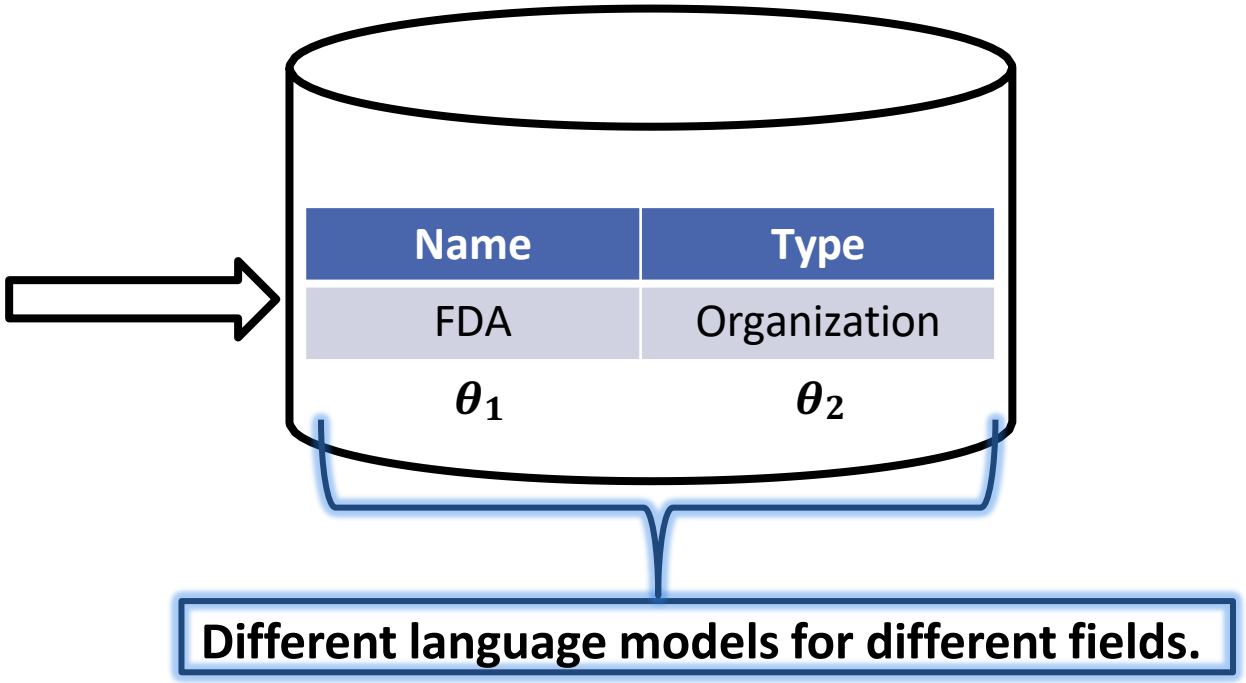
- ❑ Create a search index of all entities.
- ❑ Match the query against the entity representation.



Fielded Models: Multiple LM for Different Fields

Represent entities as "documents" with different fields.

Entity: FDA



Different language models for different fields.

Fielded Models: Multiple LM for Different Fields

Represent entities as "documents" with different fields.

Entity: FDA

Name	Type
FDA	Organization
θ_1	θ_2

Use Coordinate Ascent

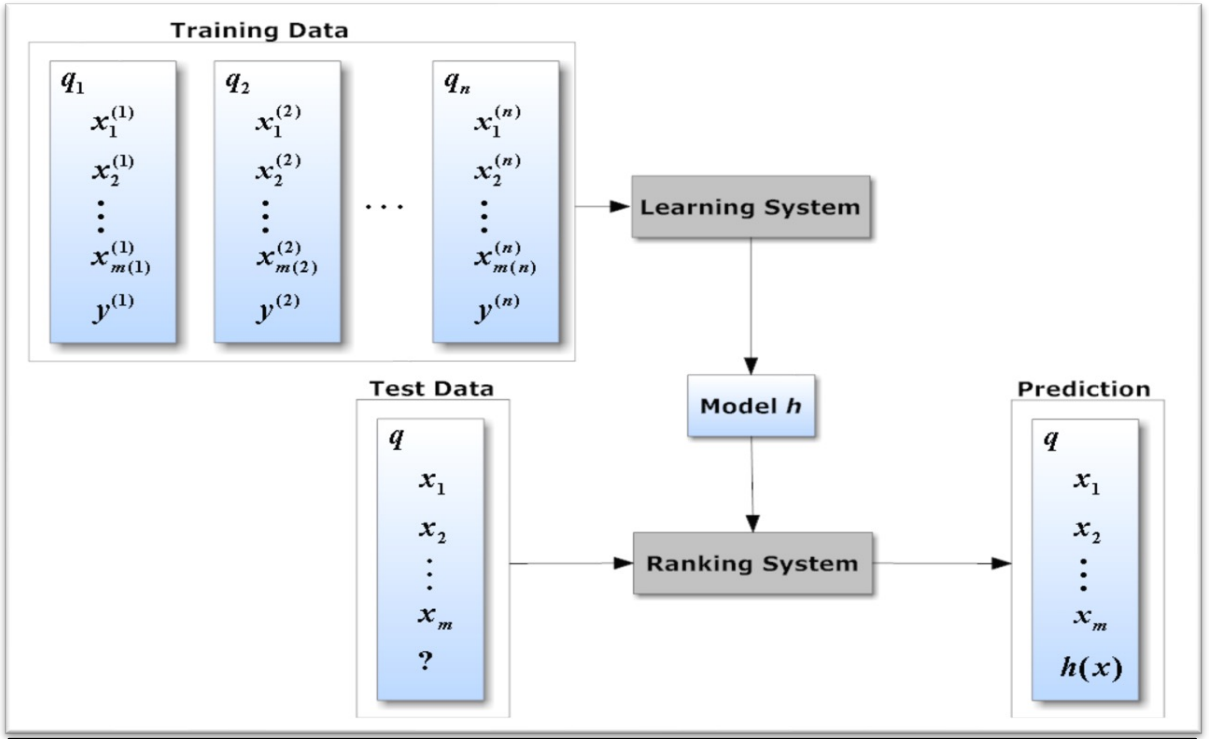
Combine using a linear mixture with field weights.

Learning-to-Rank Models

Use feature vectors of (query, entity) pairs to train a ML model.

Features:

- ❑Entity retrieval from a KB using BM25+RM3.
- ❑Whether the candidate entity is contained in the query entities.
- ❑Normalized Levenshtein Distance between the query and the mention.



Picture credit: *http://web.ist.utl.pt/~catarina.p.moreira/coursera.html*

Type-Aware Entity Retrieval

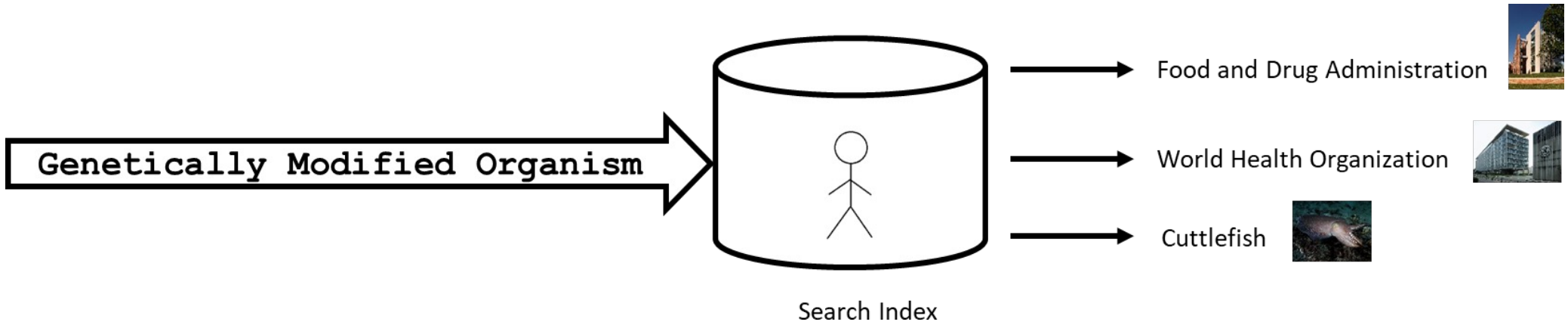
- Uses entity types from a type taxonomy (e.g., Wikipedia categories).
- Query enriched with types of entities in the query (called target types)
- Example [Balog et al., 2011]
  - Learn a probability distribution over the query and entity types.
  - Similarity = KL divergence between two distributions

How Do Systems Usually Retrieve Entities?

- ❑ Create a search index of all entities
- ❑ Match the query against the representation of the entity.

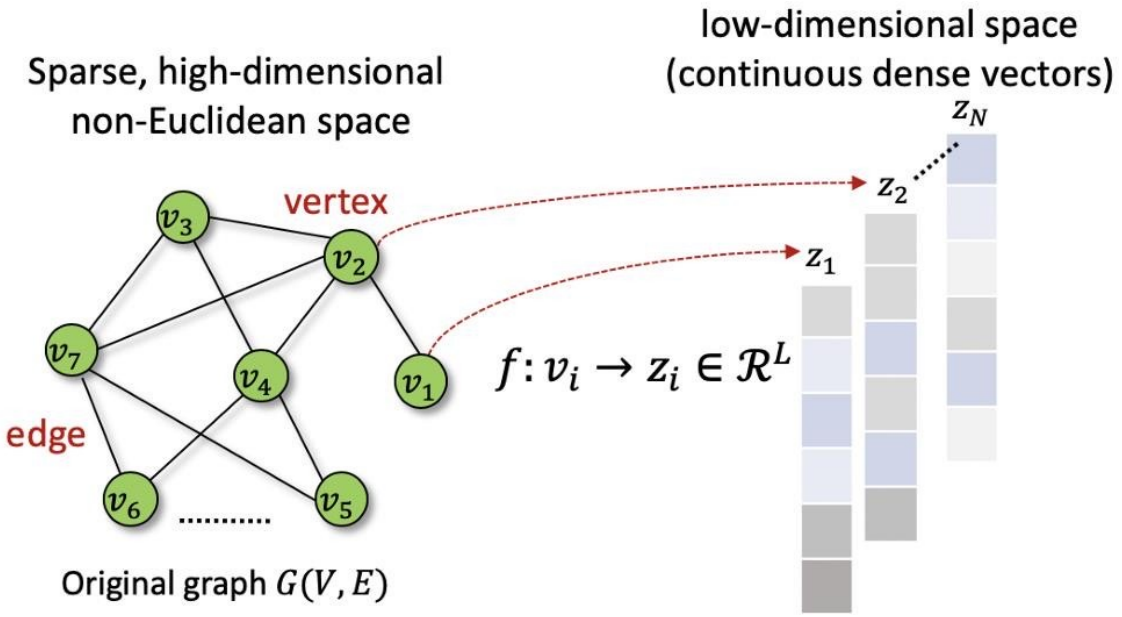
Non-neural methods: Sparse

Neural methods: Dense



Neural Entity Ranking

Graph Embeddings: Learning Dense Entity Representations



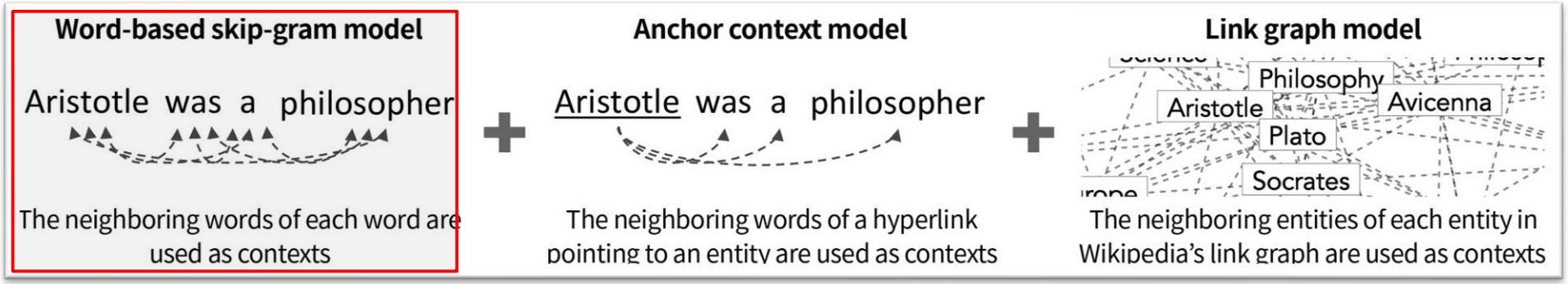
Picture Credit: *Mengjia Xu. Understanding Graph Embedding Methods and their Applications. 2020.*

- Treat the knowledge repository as a KG.
- Convert high-dimensional KG into low dimensional, dense and continuous vector spaces.
- Graph structure properties are maximally preserved.
- Examples: Wikipedia2Vec, TransE, TransR, etc.

Yamada et al., 2020

Example Graph Embedding Method: Wikipedia2Vec

- **Learning**. Jointly optimizing word-based skip-gram, anchor context, and link graph models.



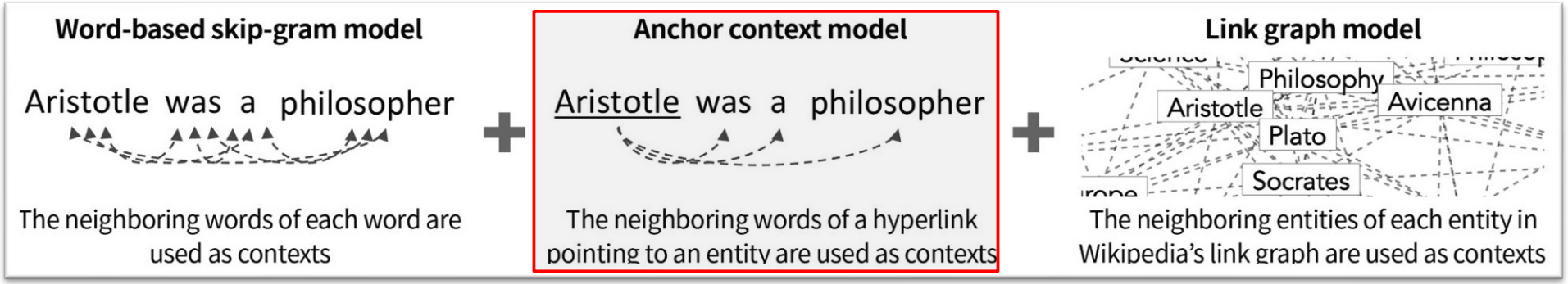
- **Word-based Skip-Gram Model.**
  - **Given**: Each word in a Wikipedia page.
  - **Do what?** Learn word embeddings.
  - **How?** Predicting the neighboring words of the given word.

Picture Credit: *Yamada et al. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. 2020*

Yamada et al., 2020

Example Graph Embedding Method: Wikipedia2Vec

- **Learning**. Jointly optimizing word-based skip-gram, anchor context, and link graph models.



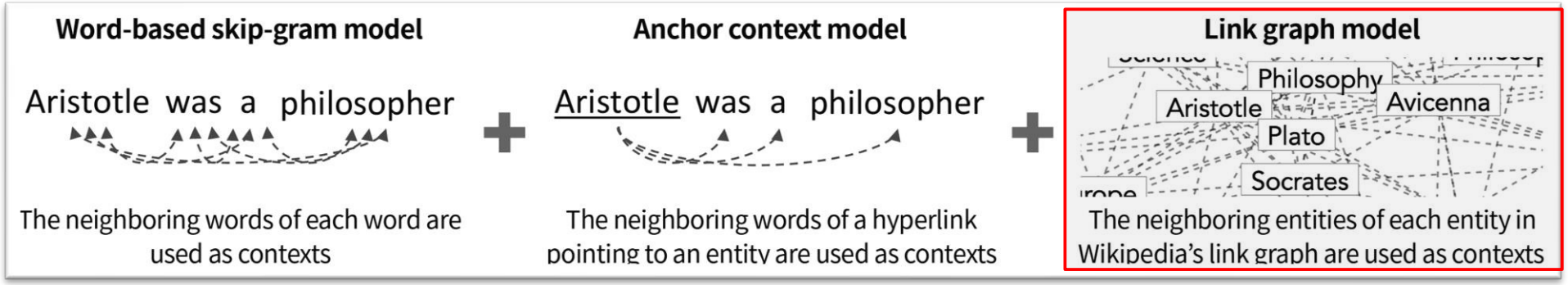
- **Anchor Context Model.**
  - **Given**: (1) Entity, and (2) surrounding words of each hyperlink in a Wikipedia page.
  - **Do what?** Place similar words and entities close together in the vector space.
  - **How?** Predicting the surrounding words given each entity.

Picture Credit: *Yamada et al. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. 2020*

Yamada et al., 2020

Example Graph Embedding Method: Wikipedia2Vec

- **Learning**. Jointly optimizing word-based skip-gram, anchor context, and link graph models.



- **Link Graph Model.**
  - **Given**: Entities on a Wikipedia page.
  - **Do what?** Learn entity embedding.
  - **How?** Predicting neighboring entities of each entity in the Wikipedia's link graph.

- Undirected graph.
- Nodes = entities.
- Edges = hyperlinks between entities.

Picture Credit: *Yamada et al. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. 2020*

Gerritse et al., 2020

GEEER: Using Wikipedia2Vec for Entity Ranking

- Re-ranks entities using Wikipedia2Vec.
- Shows that Wikipedia2Vec is useful for entity ranking.
- **General Idea**: Relevant entities for a given query are situated close (in graph embedding space) to the query entities identified by the entity linker.
- **Method**:
  - Compute the embedding-based score for an entity:

$$Score_{emb}(E, Q) = \sum_{e \in Q} C(e) \cdot \cos(\vec{E}, \vec{e})$$

  - Final Score = interpolation of the embedding-based and retrieval scores.

$$Score_{final}(E, Q) = \lambda \cdot Score_{emb}(E, Q) + (1 - \lambda) \cdot Score_{ret}(E, Q)$$

Gerritse et al., 2020

GEEER: Using Wikipedia2Vec for Entity Ranking

- Re-ranks entities using Wikipedia2Vec.
- Shows that Wikipedia2Vec is useful for entity ranking.
- **General Idea**: Relevant entities for a given query are situated close (in graph embedding space) to the query entities identified by the entity linker.
- **Method**:
  - Compute the embedding-based score for an entity:

$$Score_{emb}(E,Q) = \sum_{e \in Q} C(e) \cdot \cos(\vec{E}, \vec{e})$$

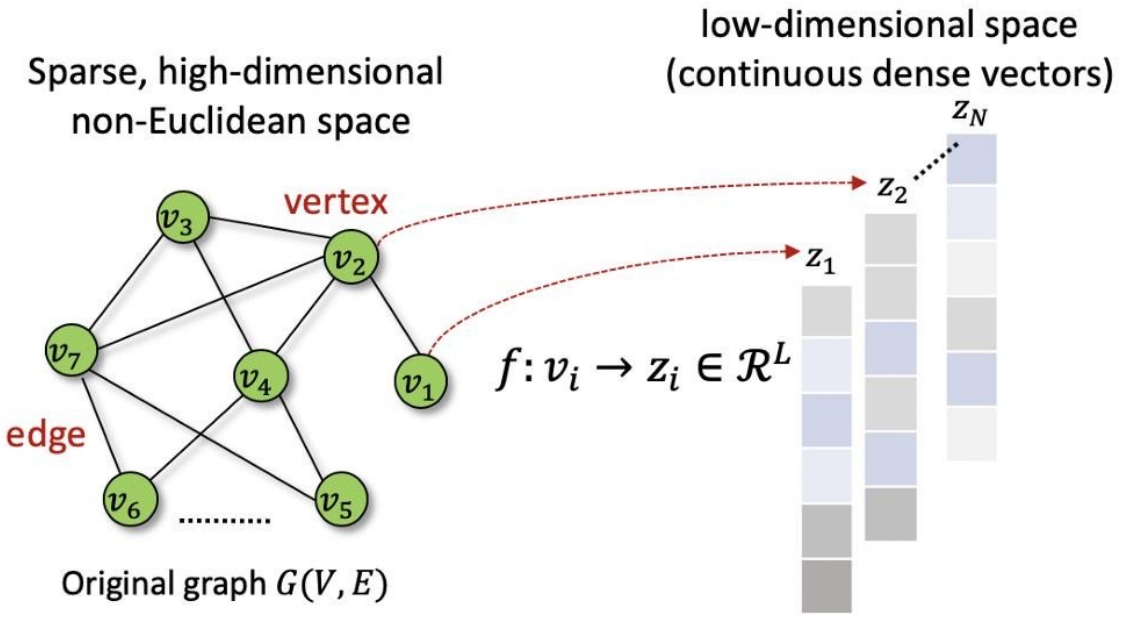
Confidence score of entity linker.

  - Final Score = interpolation of the embedding-based and retrieval scores.

$$Score_{final}(E,Q) = \lambda \cdot Score_{emb}(E,Q) + (1-\lambda) \cdot Score_{ret}(E,Q)$$

Use LTR optimized for NDCG@100.

Graph Embeddings: Issues for IR



Picture Credit: *Mengjia Xu. Understanding Graph Embedding Methods and their Applications. 2020.*

- **IR: Considers explicit query.**
- **Current graph embedding methods: Not seen the query during training (query-agnostic!)**

How About We Use BERT?

- BERT shown to be useful for document retrieval.
- **Question: Can we use BERT for entity ranking?**

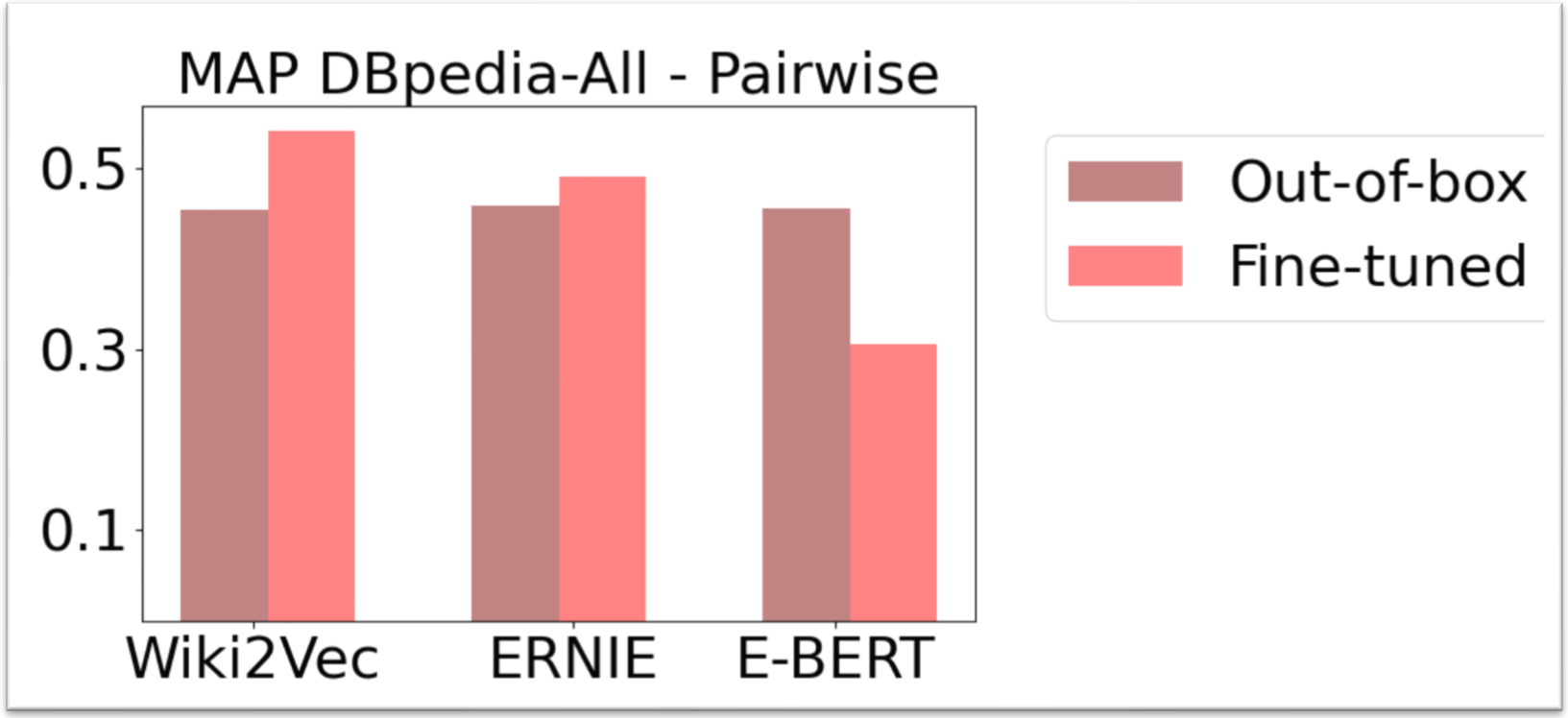
Poerner et al., 2020

One Idea: E-BERT: Inject Entity information into BERT

- **Goal:** Align Wikipedia2Vec entity vectors with BERT's native word piece vectors
- **How?** Learn a linear mapping $\boldsymbol{W}$.
- **Issue**: BERT's dictionary does not contain any entities!
- **Solution:**
  - Use common words in the vocabulary of BERT and Wikipedia2Vec.
  - Learn $\boldsymbol{W}$: Minimize squared Euclidean distance between embeddings of common words.
- **Idea:** Wikipedia2Vec embeds words and entities into the same vector space → $\boldsymbol{W}$ learnt using words can also be applied to entities.

Oza and Dietz., 2023

How Well Does BERT Understand Entities?



Results for GEEER with ERNIE and E-BERT. DBpedia-Entity v2.

Gerritse et al., 2022

EM-BERT: Entity-enriched BERT for Entity Ranking

- Aligns Wikipedia2Vec entity vectors with BERT's native word piece vectors (as in E-BERT).
- Entity vectors used with entity mentions.
- Fine-tuning: First MS MARCO passages, then DBpedia-Entity v2.

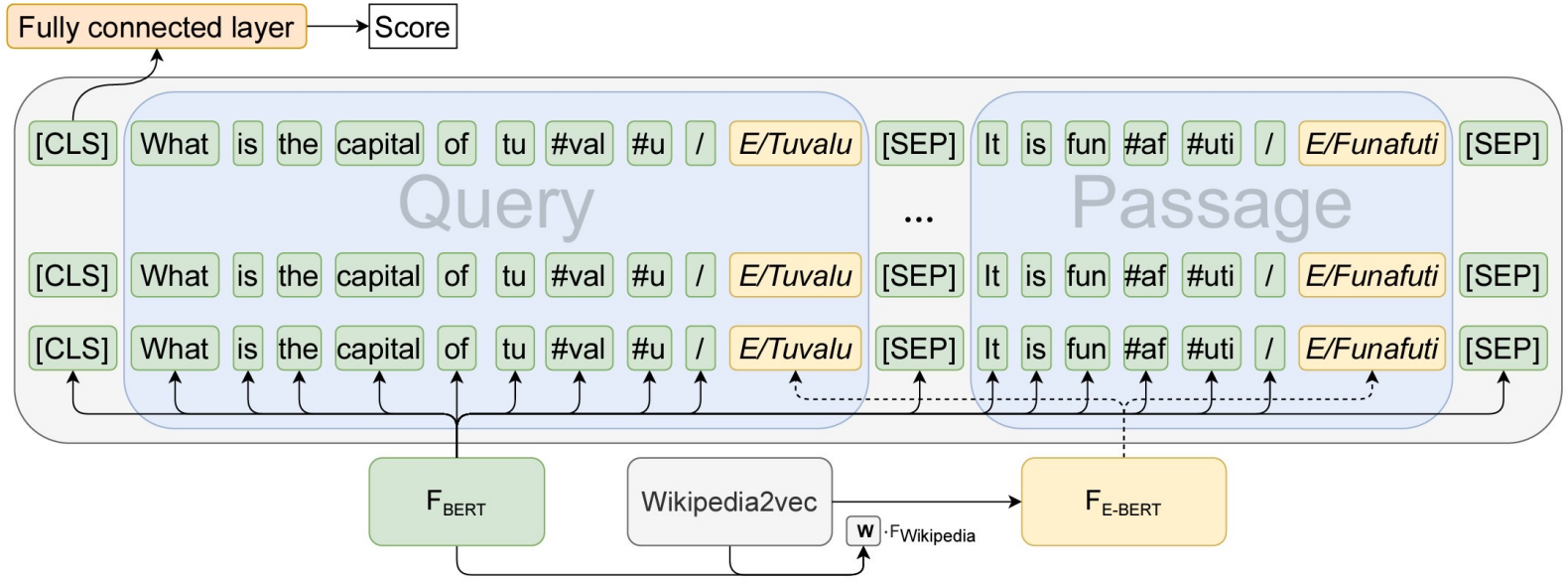


Figure 1: Illustration of the EM-BERT model. Entity annotated query and documents are tokenized and mapped to their corresponding vector representations using F_{BERT} and F_{E-BERT} functions.

Picture Credit: *Gerritse et al. Entity-aware Transformers for Entity Search. 2022.*

Gerritse et al., 2022

EM-BERT: What do we learn?

- Substantial improvements over SOTA for entity ranking!
- Helps:
  - Complex natural language queries,
  - List search queries, and
  - Queries containing tail entities

Table 2: Results on DBpedia-Entity v2 collection. Superscripts 1/2/3/4 denote statistically significant differences (better or worse) compared to base method/monoBERT (1st)/monoBERT/EM-BERT (1st), respectively. The highest value per column is marked with underline and in bold. The highest value per column and per block is marked with underline.

NDCG	SemSearch		INEX-LD		ListSearch		QALD-2		Total	
	@10	@100	@10	@100	@10	@100	@10	@100	@10	@100
ESIM$_{cg}$ [16]	0.417	0.478	0.217	0.286	0.211	0.302	0.212	0.282	0.262	0.335
KEWER [31]	-	-	-	-	-	-	-	-	0.270	0.310
BLP-TransE [10]	0.631	0.723	0.446	0.546	0.442	0.540	0.401	0.482	0.472	0.562
BM25F+KEWER [31]	0.661	0.733	0.468	0.530	0.440	0.521	0.386	0.474	0.483	0.560
BM25	0.425	0.523	0.298	0.330	0.274	0.322	0.192	0.243	0.291	0.349
+monoBERT (1st)	0.588^{1}	0.655^{1}	0.406^{1}	0.459^{1}	0.422^{1}	0.458^{1}	0.350^{1}	0.406^{1}	0.437^{1}	0.490^{1}
+monoBERT	0.577^{1}	0.640^{12}	0.409^{1}	0.452^{1}	0.423^{1}	0.449^{12}	0.347^{1}	0.392^{12}	0.435^{1}	0.479^{12}
+EM-BERT (1st)	0.593^{1}	0.667^{13}	0.395^{1}	0.448^{1}	0.436^{1}	0.465^{1}	0.351^{1}	0.403^{1}	0.440^{1}	0.492^{13}
+EM-BERT	0.612^{1}	0.672^{1}	0.392^{1}	0.434^{12}	0.478^{1234}	0.469^{13}	0.375^{1234}	0.418^{134}	0.461^{1234}	0.495^{13}

One Idea: Injecting Entity information into BERT

Another Idea: Can we utilize existing knowledge in BERT?

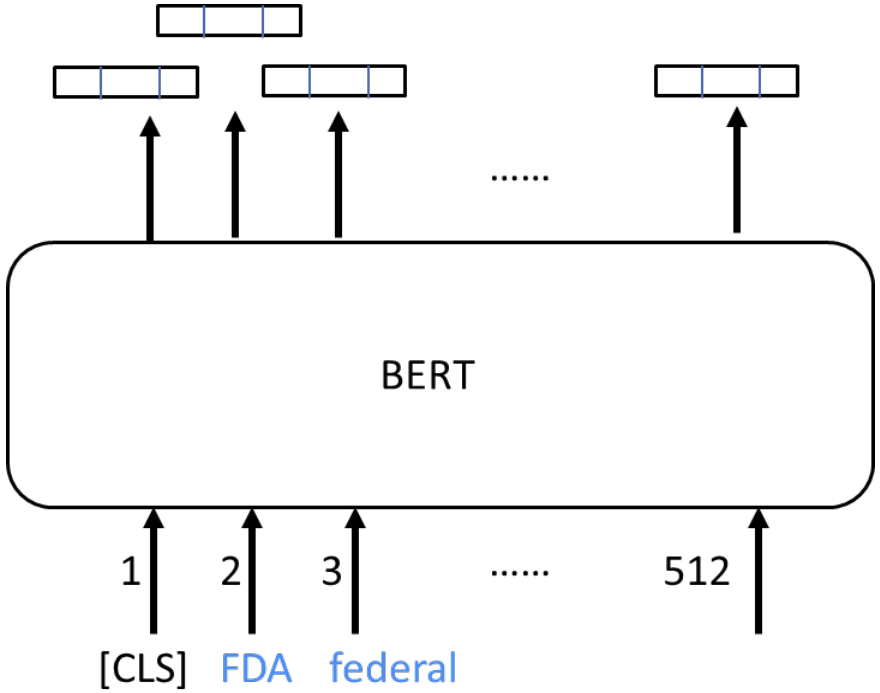
- BERT has already seen a lot of the world (from books and Wikipedia).
- BERT can probably infer the connection between the query and entity from a term-based entity description.
- Term-based entity description = Introductory Wikipedia paragraph (most often).

Query: Genetically Modified Organism
Relevant Entity: Food and Drug Administration

Lead Text

Food and Drug Administration

The United States **Food and Drug Administration** (**FDA** or **USFDA**) is a federal agency of the Department of Health and Human Services. The FDA is responsible for protecting and promoting public health through the control and supervision of food safety, tobacco products, dietary supplements, prescription and over-the-counter pharmaceutical drugs (medications), vaccines, biopharmaceuticals, blood transfusions, medical devices, electromagnetic radiation emitting devices (ERED), cosmetics, animal foods & feed[3] and veterinary products.



- Lead Text = Static entity description
- No knowledge of the query!
- Static entity embeddings.

Query: Genetically Modified Organism
Relevant Entity: Food and Drug Administration

FDA regulates most human and animal food, including GMO foods. In doing so, FDA makes sure that foods that are GMOs or have GMO ingredients meet the same strict safety standards as all other foods. FDA sets and enforces food safety standards […]

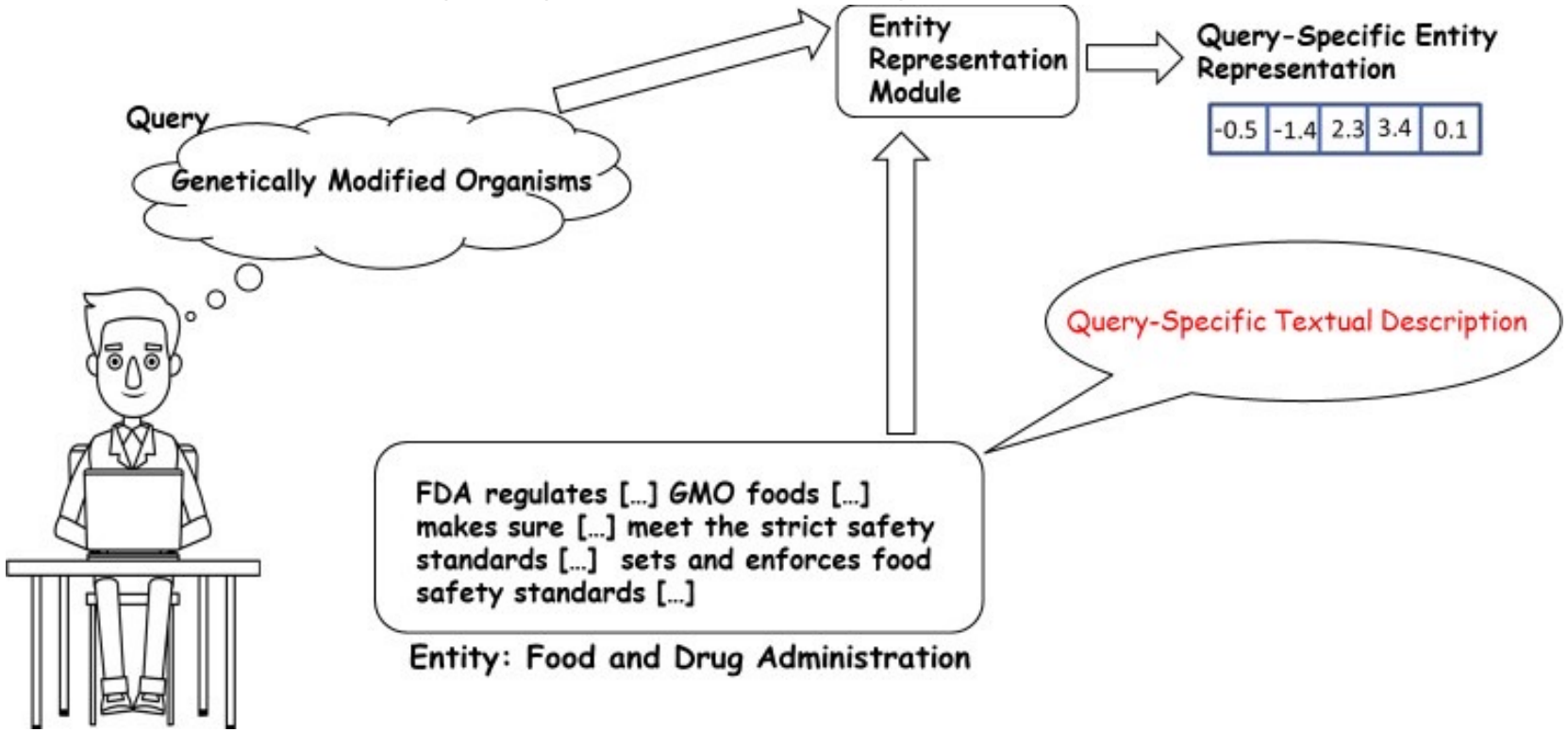
Clarifies the connection between the query and entity.

What if we had this text instead of lead text?

Chatterjee and Dietz., 2022

BERT-ER: Query-Specific Entity Descriptions

Query-specific entity descriptions→ descriptions that mention relevant connections between the query and the entity.



Food and Drug Administration

Chatterjee and Dietz., 2022

The United States **Food and Drug Administration** (**FDA** or **USFDA**) is a federal agency of the Department of Health and Human Services. The FDA is responsible for protecting and promoting public health through the control and supervision of food safety, tobacco products, dietary supplements, prescription and over-the-counter pharmaceutical drugs (medications), vaccines, biopharmaceuticals, blood transfusions, medical devices, electromagnetic radiation emitting devices (ERED), cosmetics, animal foods & feed[3] and veterinary products.

1 Organizational structure
2 Location
3 Scope and funding
4 Regulatory programs
5 Science and research programs
6 Data management
7 History

Aspects = Top-Level Sections

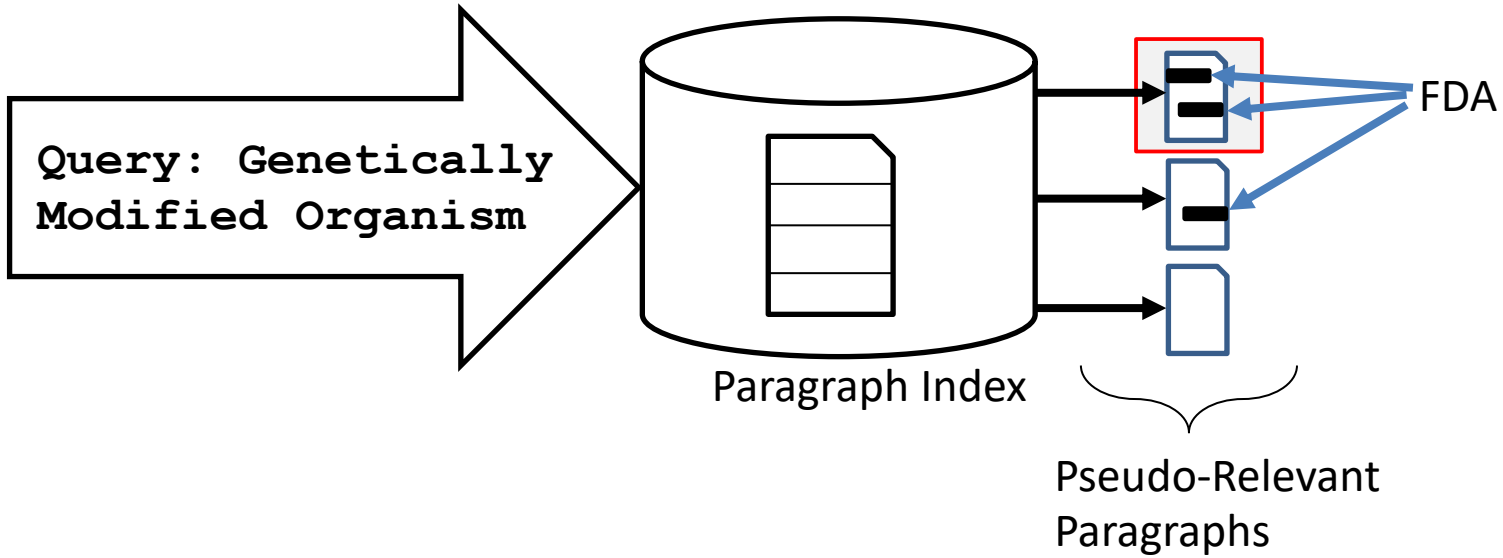
Query-Specific Entity Descriptions: Alternative 1

❑**Using Wikipedia: Top-Level Sections**
❑Identify relevant top-level sections from the Wikipedia page. ***(Why? —because the lead text is does not elaborate the relevance!)***
❑Use catalog of top-level sections (aspects) from Ramsdell et al., 2020.
❑ **Downside:** ***Wikipedia articles often do not contain all relevant information!***

Query-Specific Entity Descriptions: Alternative 2

Chatterjee and Dietz., 2022

❑ **Using Paragraph Collection: PRF Passages**

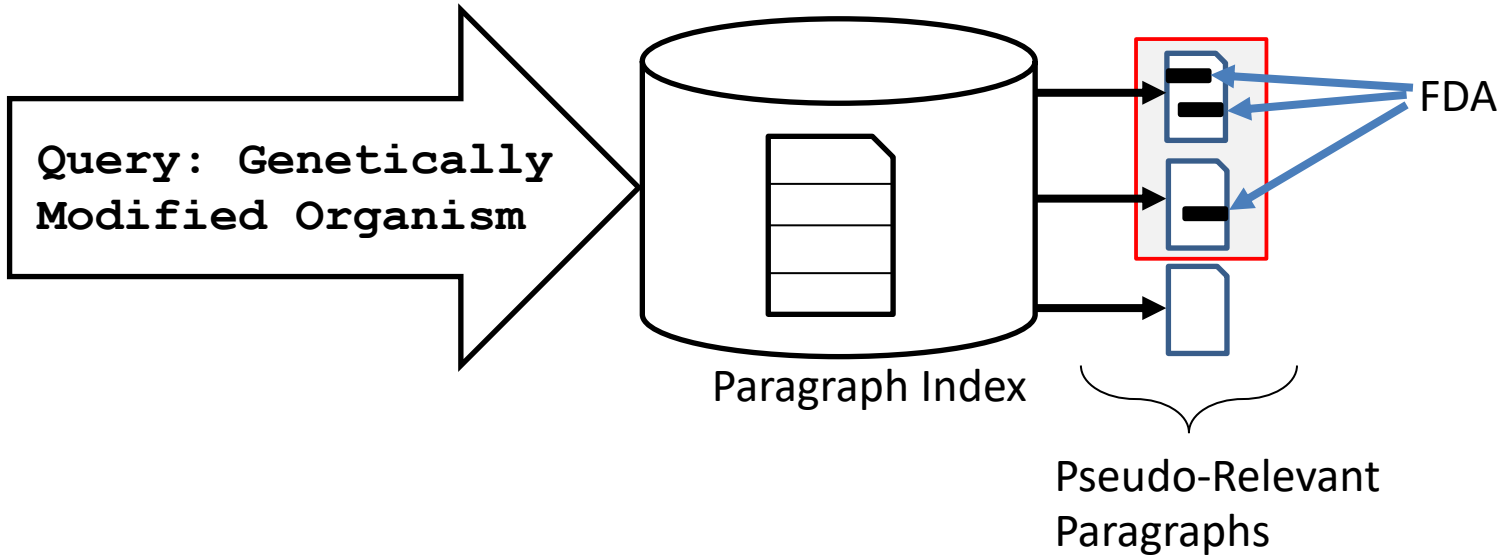


❑ **Downside*: Entity may not be central to the discussion in the text!***

Query-Specific Entity Descriptions: Alternative 3

Chatterjee and Dietz., 2022

❑ **Using Paragraph Collection: Entity-Support Passages**



❑ Re-rank these documents.
❑ Two criteria:
1. How many relevant connections between query and entity?
2. Are the relevant connections central to the discussion in the text?

Entity-Support Passages:
Chatterjee and Dietz, ICTIR 2019.

BERT-ER: Alternative Approaches for Query-Specific Entity Descriptions

Chatterjee and Dietz, SIGIR 2022

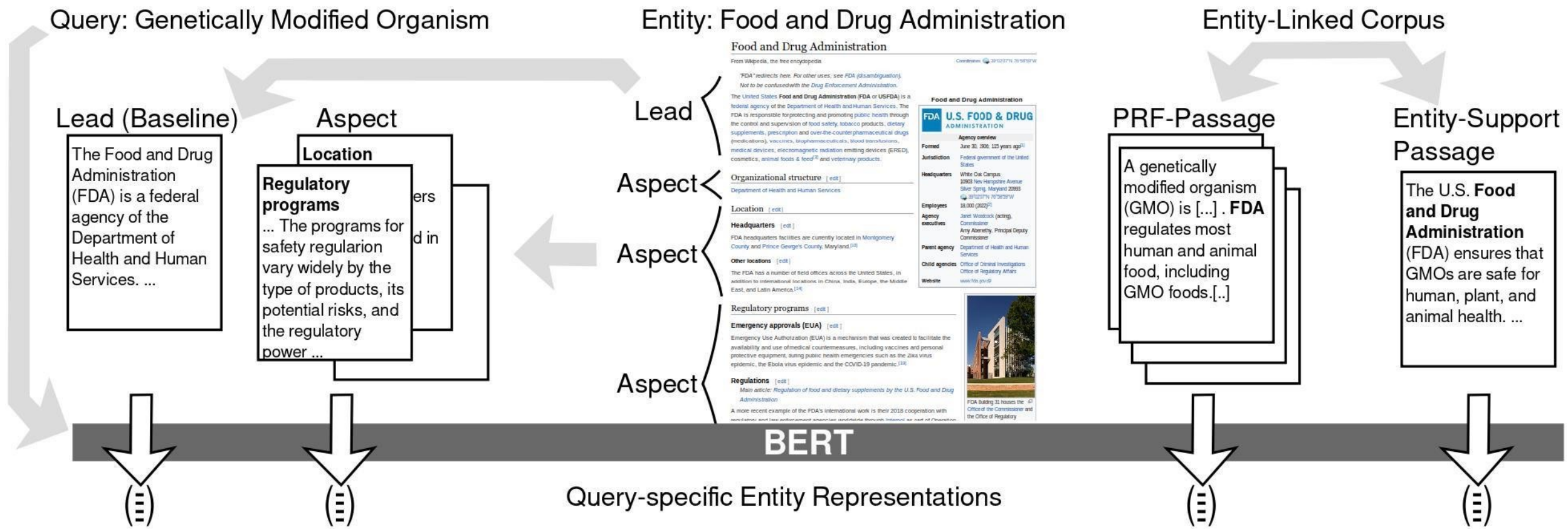


Table 3: Results on DBpedia-Entity v2 (separated by different subsets). Trained using 5-fold cross-validation. ▲ denotes significant improvement and ▼ denotes significant deterioration compared to ⋆. Only best baselines shown.

	All			SemSearch_ES			ListSearch			INEX_LD			QALD2		
	MAP	P@R	NDCG@100	MAP	P@R	NDCG@100	MAP	P@R	NDCG@100	MAP	P@R	NDCG@100	MAP	P@R	NDCG@100
BERT-LeadText++⋆	0.45⋆	0.41⋆	0.68⋆	0.60⋆	0.54⋆	0.77⋆	0.43⋆	0.40⋆	0.69⋆	0.43⋆	0.40⋆	0.69⋆	0.34⋆	0.32⋆	0.60⋆
BM25F-CA [23]	0.45	0.43▲	0.68	0.61	0.55	0.78	0.44	0.43▲	0.68	0.42▼	0.41▲	0.67▼	0.37▲	0.36▲	0.46▼
ENT-Rank [14]	0.48▲	0.44▲	0.71▲	0.59	0.50▼	0.78	0.49▲	0.47▲	0.74▲	0.43	0.42▲	0.70▲	0.40▲	0.37	0.64▲
GEEER [20]	0.37▼	0.38▼	0.57▼	0.56▼	0.53▼	0.72▼	0.34▼	0.38▼	0.54▼	0.34▼	0.35▼	0.55▼	0.27▼	0.29▼	0.48▼
LTR-ASP [9]	0.43▼	0.39▼	0.68	0.55▼	0.47▼	0.74▼	0.43	0.41▲	0.69	0.41▼	0.38▼	0.67▼	0.36▲	0.32	0.62▲
BERT-ER++	**0.50▲**	**0.46▲**	**0.72▲**	**0.63▲**	**0.57▲**	**0.81▲**	**0.51▲**	**0.47▲**	**0.74▲**	**0.47▲**	**0.44▲**	**0.71▲**	**0.41▲**	**0.38▲**	**0.65▲**

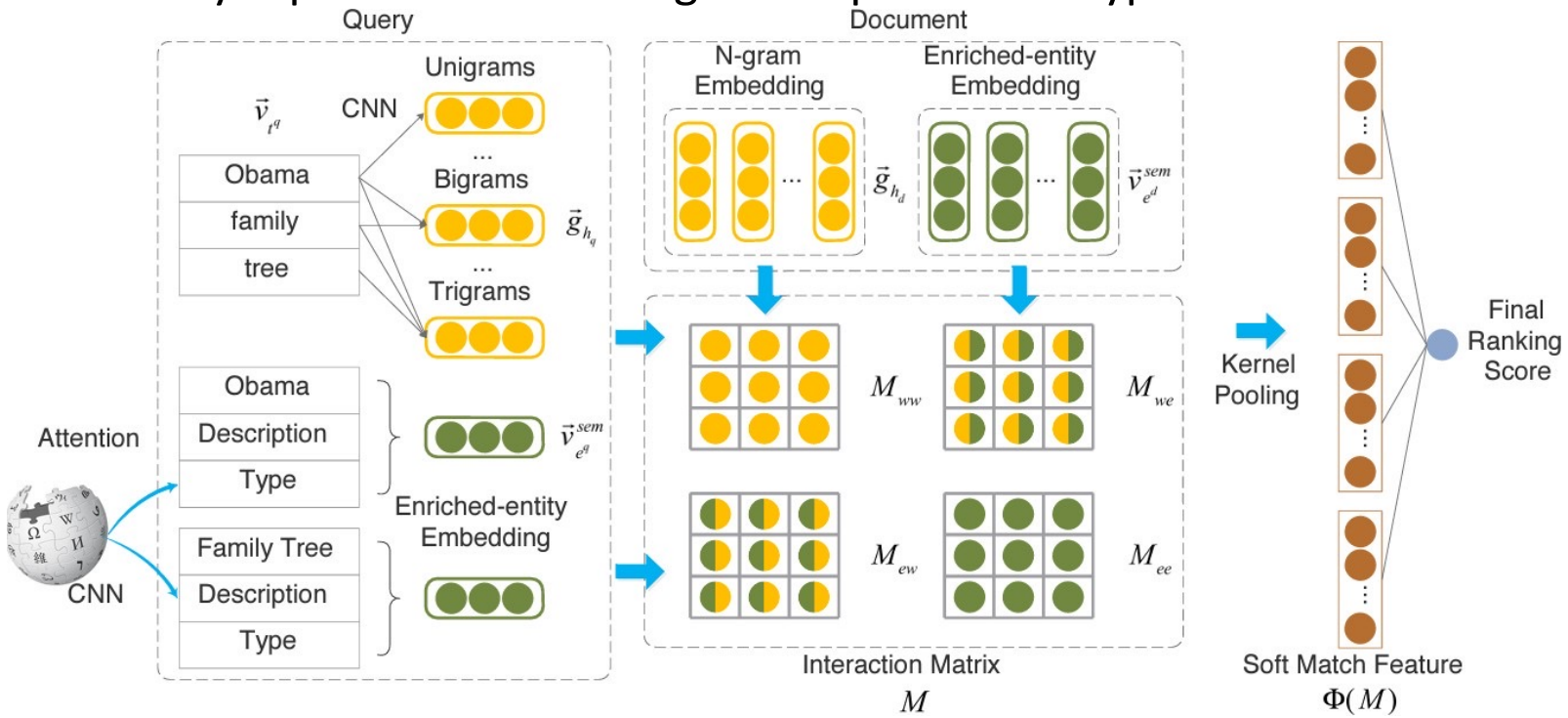
Results for BERT-ER

Entity-Centric Document Retrieval

Entity-Duet Neural Ranking Model

Liu et al., 2018

- ❑ Incorporates entities in interaction-based neural ranking models.
- ❑ Learns entity representations using: descriptions and types.



***Picture Credits**: Liu et al. Entity-Duet Neural Ranking. 2018.*

Dense Retrieval With Entity Views

Tran and Yates, 2022

- Enrich query/document representation with entity representations.
- Cluster entities → Entity embeddings using clusters → Clusters act as "views" of the document.

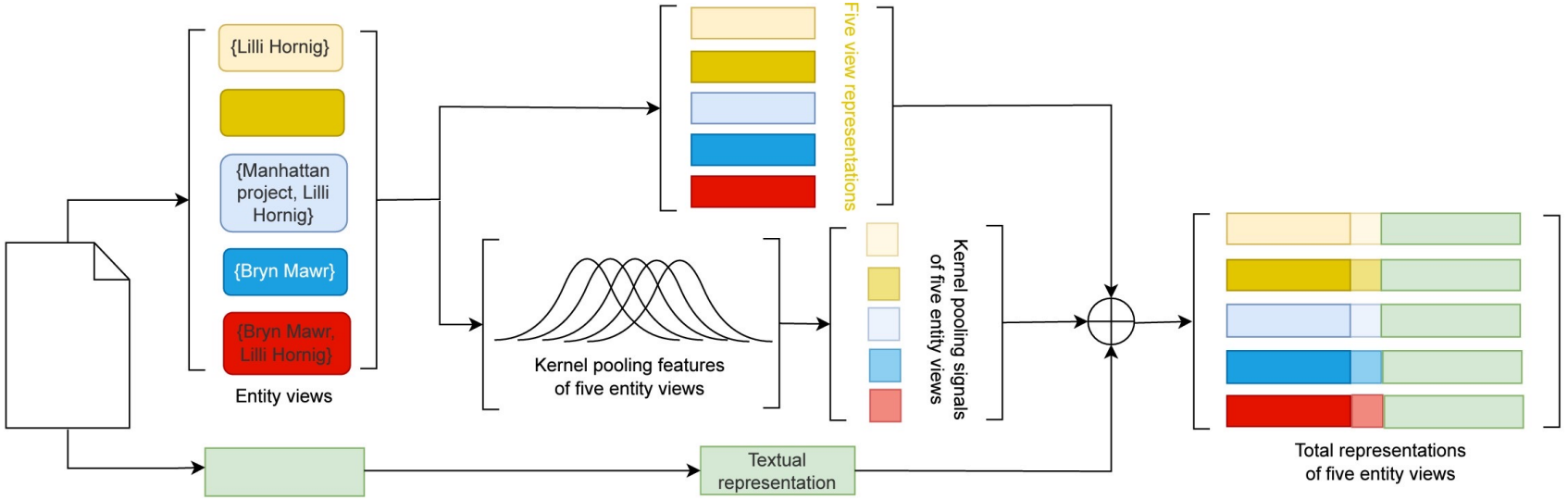


Figure 1: Overview of EVA with multiple representations. Entity clusters such as {Lilli Hornig}, {Manhattan project}, {Manhattan project, Lilli Hornig}, {Bryn Mawr} and {Bryn Mawr, Lilli Hornig} can be understood as different entity views of the passage. EVA generates one total representation for each view, which enriches a textual representation with the entities present.

***Picture Credits**: Tran and Yates. Dense Retrieval with Entity Views. 2022.*

Methods	Latency	TREC DL 19			TREC DL 20			DL HARD			MS MARCO Dev		
	(ms)	nDCG	MRR	MAP	nDCG	MRR	MAP	nDCG	MRR	MAP	nDCG	MRR	MAP
Low latency (<100 ms)													
BM25	13	0.506	0.702	0.301	0.480	0.653	0.286	0.285	0.465	0.159	0.228	0.184	0.193
ANCE	25	0.621	0.763	0.361	0.605	0.786	0.373	0.335	0.446	0.193	0.368	0.311	0.317
ERNIE Tuned	29	0.574	0.728	0.326	0.573	0.760	0.348	0.287	0.388	0.163	0.320	0.267	0.274
ERNIE Multi	70	0.669†	0.822	0.422†	0.631†	0.891†	0.394†	0.329	0.452	0.198	0.344†	0.291†	0.296†
TAS BERT	28	0.693	0.835	0.442	0.673	0.812	0.451	0.360	0.472	0.224	0.395	0.334	0.340
EVA Single	40	0.672	0.853	0.429	0.642	0.813	0.428	0.363	0.481	0.224	0.374	0.316	0.322
EVA Multi	76	0.733	0.853	**0.483**	**0.694**	**0.855**†	**0.456**	0.397†	0.521†	0.240	**0.407**†	0.346†	0.350†
EVA Multi-KNRM	74	**0.743**†	**0.879**	0.482	0.680	0.827	0.440	**0.402**†	**0.532**†	**0.253**	0.406†	**0.347**†	**0.351**†

Results for Dense Retrieval with Entity Views