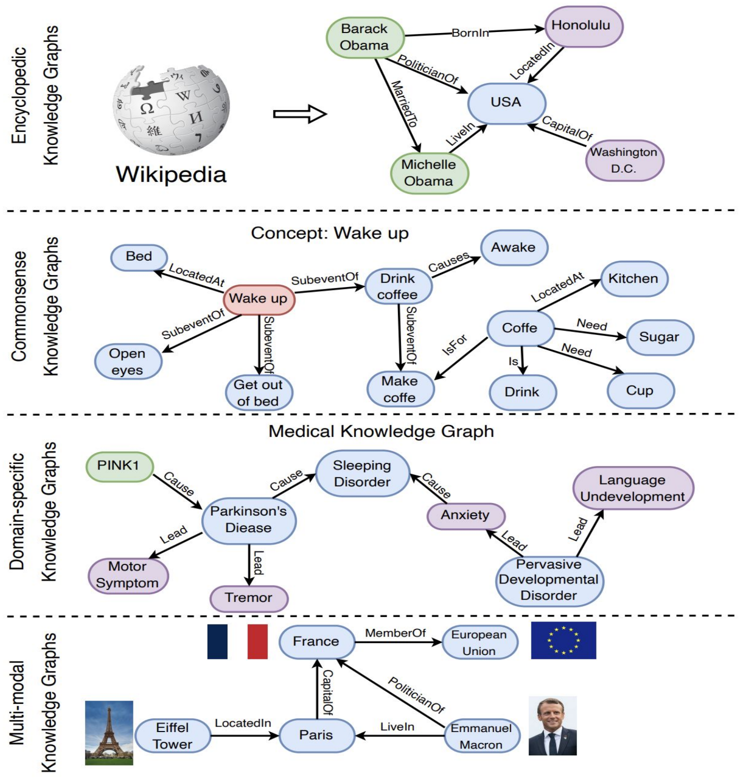
Optimizing KGQA Systems: A Systematic Comparative Study of Negative Sampling Strategies in KGE Models

Medhasweta Sen, Nina Ebensperger, Nayaeun Kwon

2024 Capstone Project

Knowledge Graphs

- Encyclopedic Knowledge Graphs
- Commonsense Knowledge Graphs
- Domain-specific Knowledge Graphs
- Multi-modal Knowledge Graphs



Knowledge Graph Embedding (KGEs)

Why do we need KGE?

How do we compute KGEs?

ComplEx Embeddings:

Scoring Function:

$$
\begin{aligned}
\phi(r, s, o; \Theta) &= \mathrm{Re}(< w_r, e_s, \bar{e}_o >) \\
&= \mathrm{Re}(\sum_{k=1}^{K} w_{rk} e_{sk} \bar{e}_{ok}) \\
&= \langle \mathrm{Re}(w_r), \mathrm{Re}(e_s), \mathrm{Re}(e_o) \rangle \\
&\quad + \langle \mathrm{Re}(w_r), \mathrm{Im}(e_s), \mathrm{Im}(e_o) \rangle \\
&\quad + \langle \mathrm{Im}(w_r), \mathrm{Re}(e_s), \mathrm{Im}(e_o) \rangle \\
&\quad - \langle \mathrm{Im}(w_r), \mathrm{Im}(e_s), \mathrm{Re}(e_o) \rangle
\end{aligned}
$$

Properties:

✅ **Symmetry**
✅ **1-to-N/N-to-1**
❌ **Composition**
✅ **Antisymmetry**
✅ **Inverse**

DistMult Embeddings:

Scoring Function:

$$L(\Omega) = \sum_{(e_1,r,e_2)\in T} \sum_{(e'_1,r,e'_2)\in T'} \max\{S_{(e'_1,r,e'_2)} - S_{(e_1,r,e_2)} + 1, 0\}$$

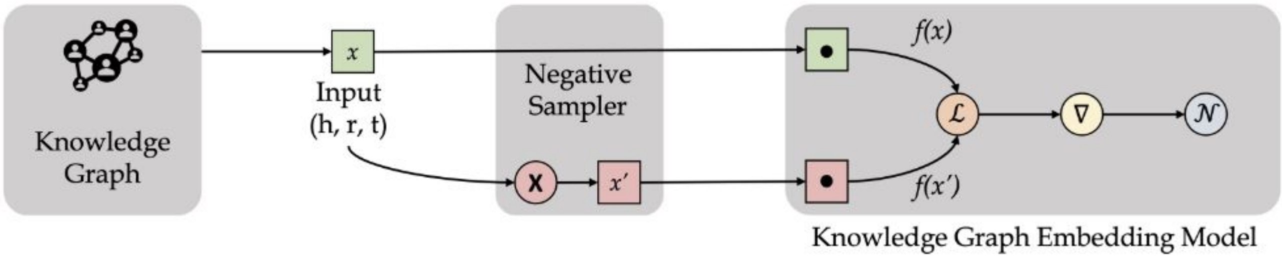
Properties:

✅ **Symmetry**
✅ **1-to-N/N-to-1**
❌ **Composition**
❌ **Antisymmetry**
❌ **Inverse**

Negative Sampling in KGE Generation

Methods:

- Uniform Sampling: Random replacement of head or tail entities.
- Random Corrupt Sampling: Extends uniform sampling by also corrupting relations.
- Batch NS: Reuses samples within the same mini-batch for efficiency.

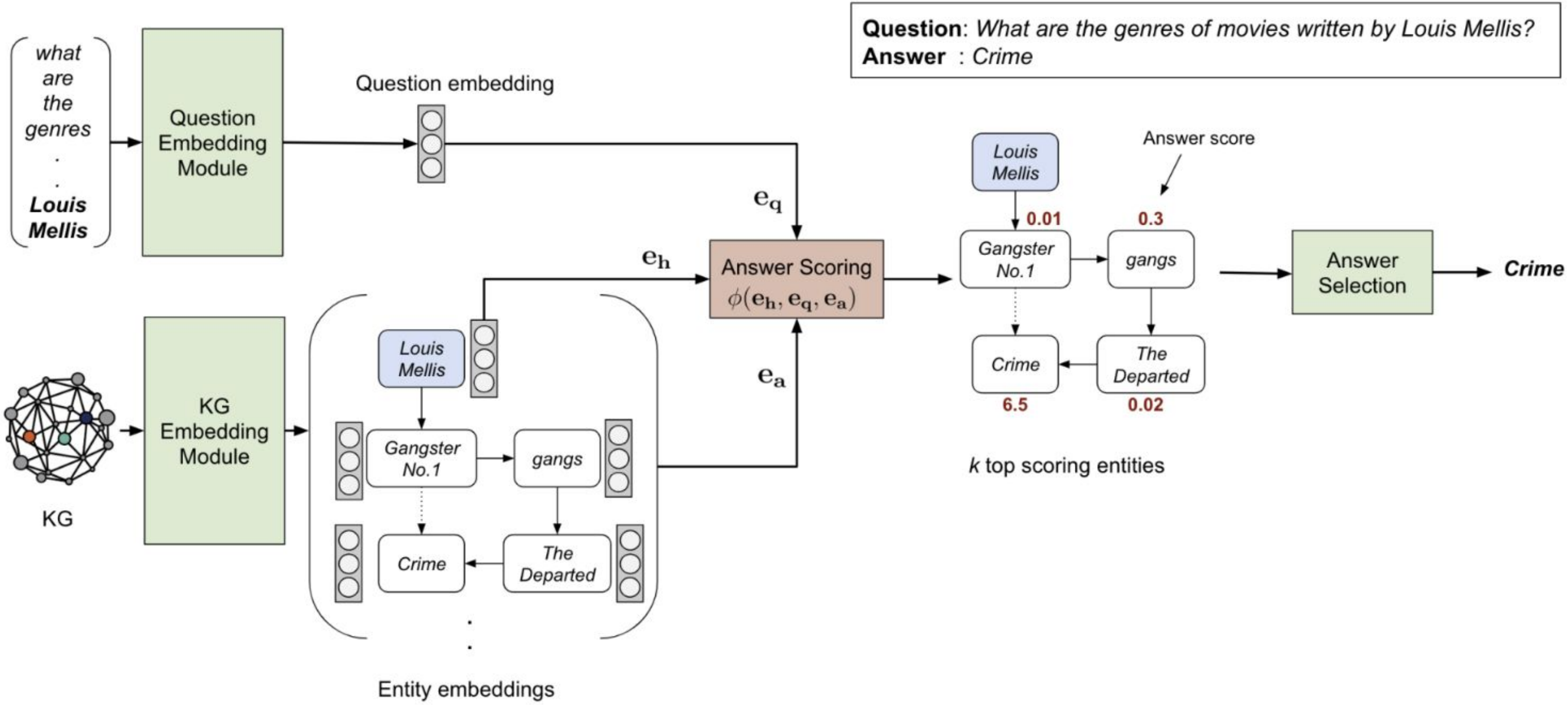


Performance of KGEs with varying Negative Sampling Techniques

KGE	Sampling	Mean Rank	MRR	Hits@10
ComplEx	Uniform	397.48	0.2742	0.3664
	Random Corrupt	**357.62**	**0.2849**	**0.3886**
	Batch NS	463.07	0.2668	0.3459
DistMult	Uniform	528.17	0.2386	0.3360
	Random Corrupt	477.78	0.2554	0.3400
	Batch NS	660.01	0.1922	0.2756

Why do we need Embed KGQA?

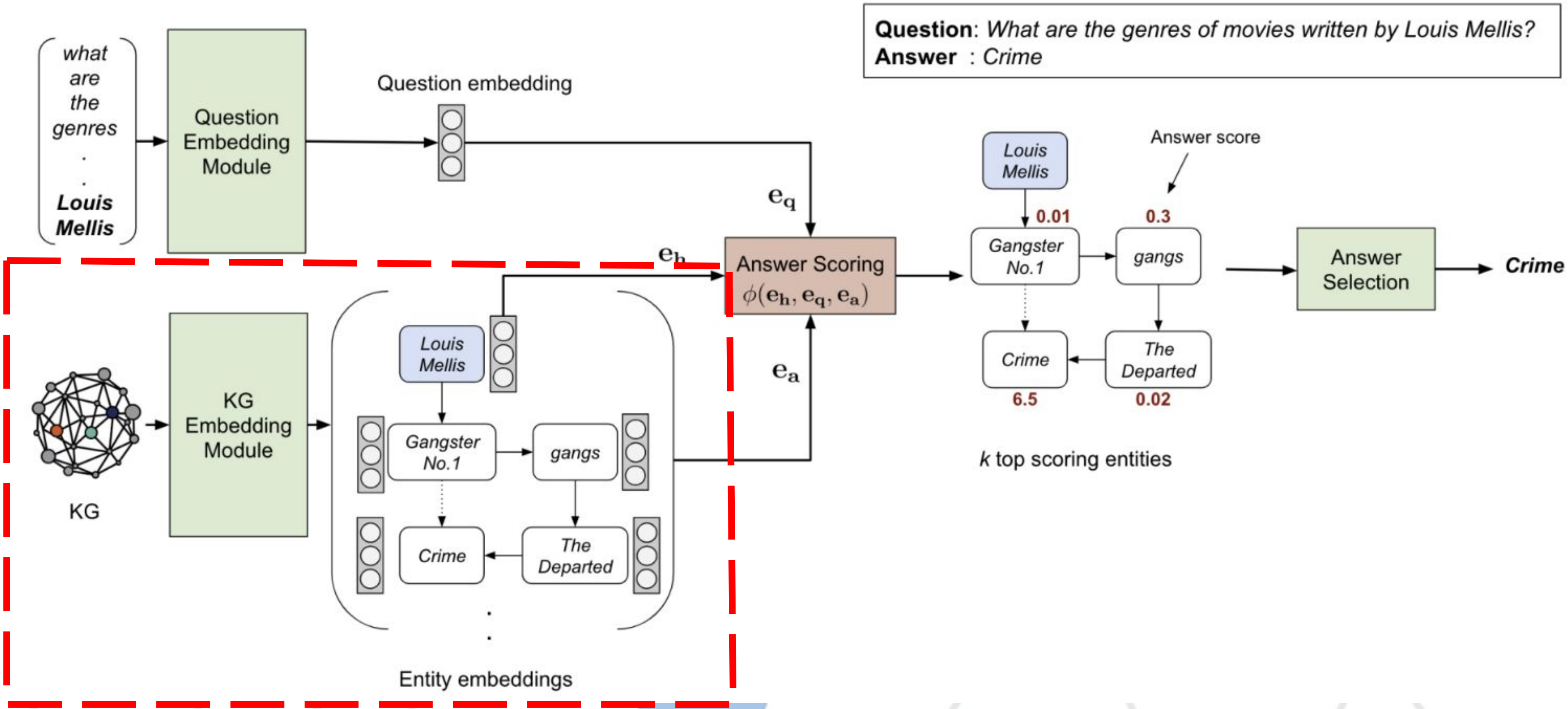
EmbedKGQA



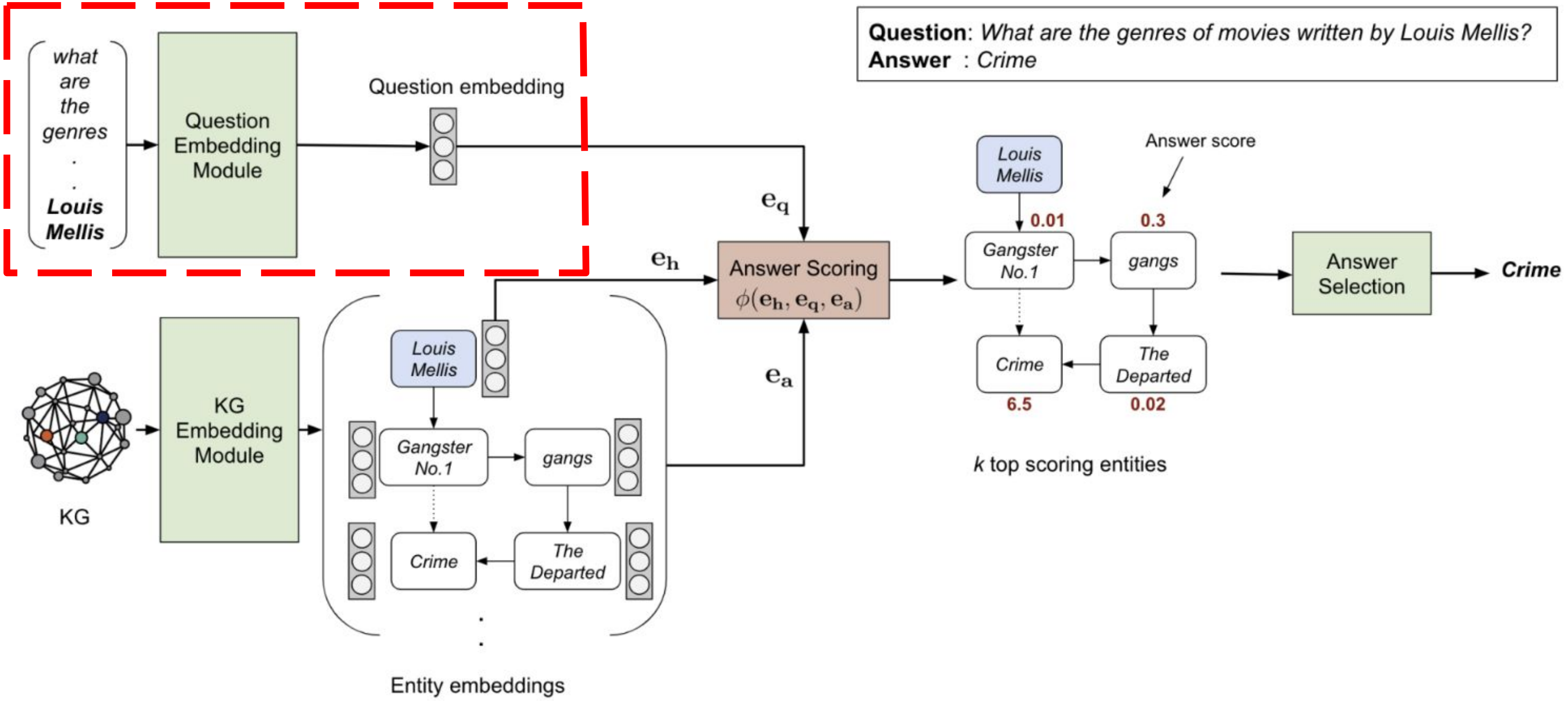
Question: What are the genres of movies written by Louis Mellis?
Answer : Crime
what are the genres . . Louis Mellis
Question Embedding Module
Question embedding
$\mathbf{e_q}$
$\mathbf{e_h}$
Answer Scoring
$\phi(\mathbf{e_h}, \mathbf{e_q}, \mathbf{e_a})$
$\mathbf{e_a}$
KG
KG Embedding Module
Louis Mellis
Gangster No.1
gangs
Crime
The Departed
Entity embeddings
Answer score
0.01
0.3
6.5
0.02
k top scoring entities
Answer Selection
Crime

How do we compute KGEs?

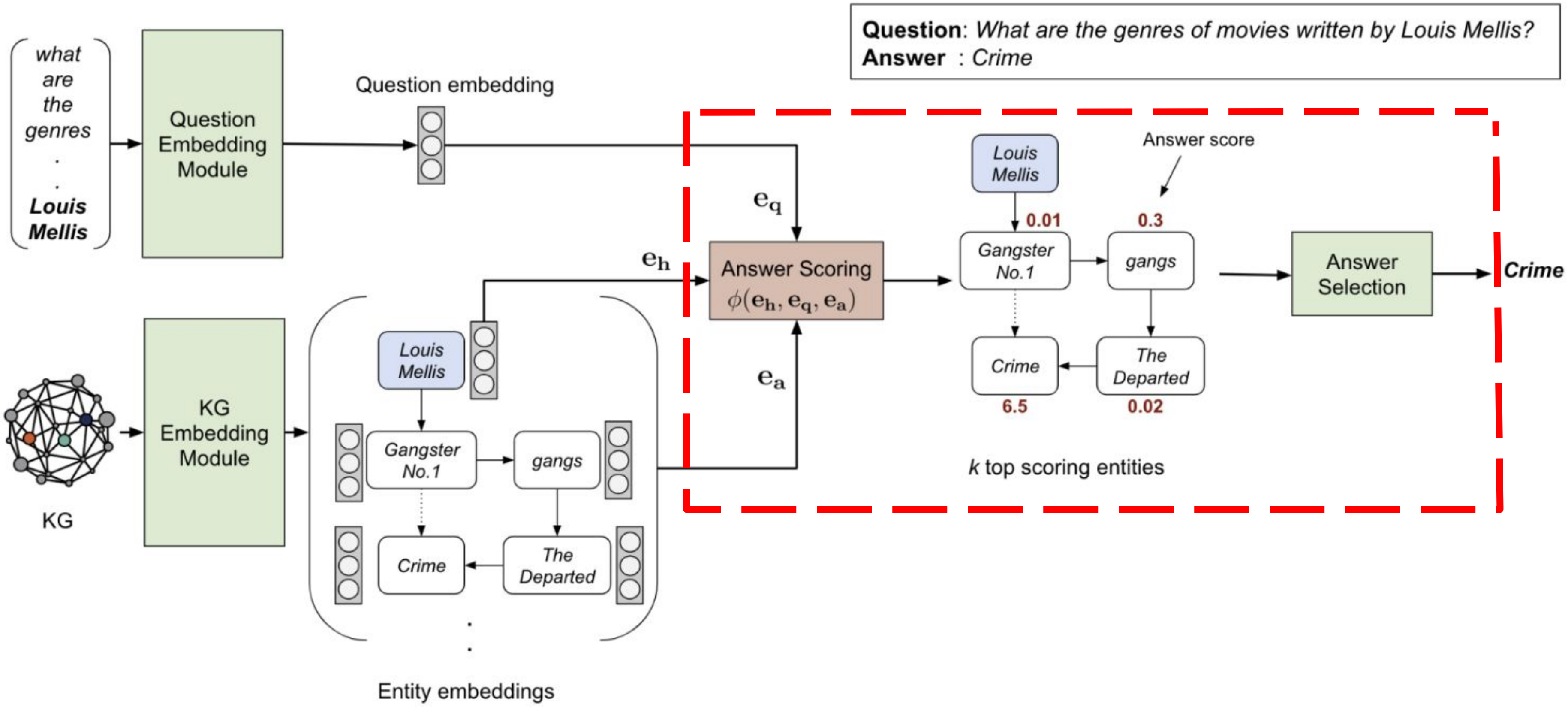
EmbedKGQA: KG Embedding Module



EmbedKGQA: Question Embedding Module



EmbedKGQA: Answer Scoring in EmbedKGQA



Experiment: Research Design

- Knowledge Graph Embedding: Create impactful knowledge graph embedding using **ComplEx** and **DistMult**
- Negative Sampling Techniques: Application of **Uniform**, **Random Corrupt**, and **Batch Negative Sampling** to test different KGE setups.
- Question Embedding: Use of **RoBERTa** and **SentenceTransformer** with KGEs in the Embed-KGQA pipeline.
- Answer Selection Module: Utilization of a **scoring function** to calculate compatibility scores between question embeddings and potential answer entities.

Experiment: Dataset

- Utilization of the MetaQA dataset tailored for the movie domain with over 400k questions.
- Includes about 135k triples, 43k entities, and 9 distinct relations, enhancing the breadth of KGQA evaluation.

Experiment: Classical Method (Cosine Similarity)

- Purpose: To measure the effectiveness of the EmbedKGQA system using cosine similarity.
- Process Overview:
  - Vectorization: Transform questions and KG into vector forms using KG and Question Embedding Modules. (SentenceTransformer: all-MiniLM-L6-v2)
  - Cosine Similarity Calculation: Determine the cosine of the angle between question vectors and candidate answer vectors.
  - Answer Selection: Choose the answer with the highest similarity score.

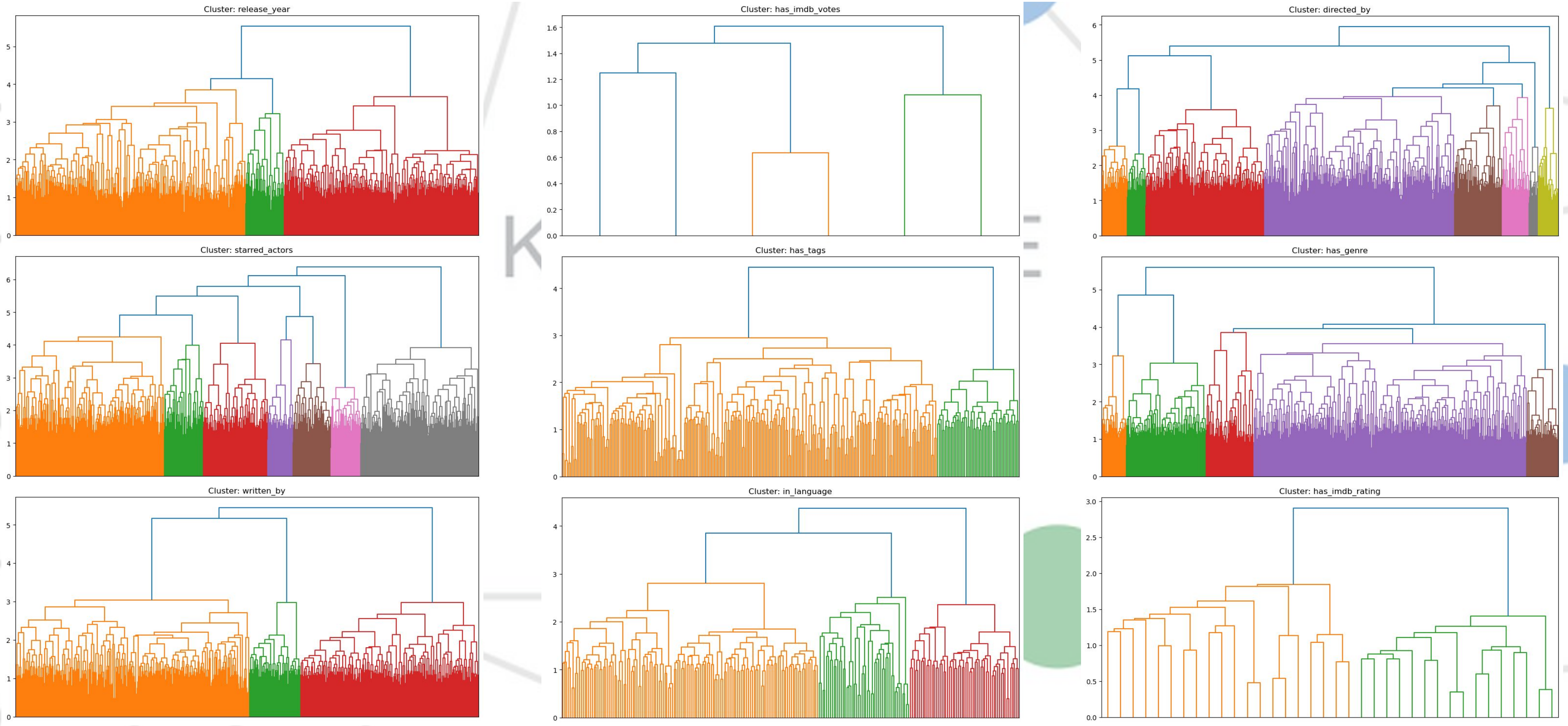
Method	Success Rate (%)
Cosine Similarity (Benchmark)	5.73%
EmbedKGQA	36.12%

Results: Question Answering

Table 2: Evaluation Metrics for Different Question Answering Pipeline Performances

Question Encoder	KGE	Sampling	Accuracy	Hits@5	Hits@10
RoBERTa	ComplEx	Uniform	0.357495	0.791998	0.965316
		Random Corrupt	**0.392279**	**0.824068**	**0.967327**
		Batch NS	0.346570	0.761978	0.916406
	DistMult	Uniform	0.264101	0.606211	0.808269
		Random Corrupt	0.287351	0.596211	0.798269
		Batch NS	0.224916	0.555009	0.729066
SentenceTransformer	ComplEx	Uniform	0.274033	0.565004	0.719332
		Random Corrupt	0.277615	0.594501	0.722047
		Batch NS	0.262626	0.556406	0.699058
	DistMult	Uniform	0.141218	0.398723	0.527073
		Random Corrupt	0.175541	0.445541	0.623555
		Batch NS	0.139007	0.396813	0.527576

Hierarchical Analysis by Relations



Conclusion

- Key Findings: The study highlights that the **ComplEx** model, enhanced by **Random Corrupt** Negative Sampling, outperforms others in terms of Mean Rank, Mean Reciprocal Rank (MRR), and Hits@k.
- Impact on KGQA: This combination significantly boosts KGQA performance by effectively capturing complex relational dynamics within knowledge graphs.
- Role of Advanced NLP: Incorporating advanced NLP techniques like **RoBERTa** into the EmbedKGQA pipeline critically enhances question answering accuracy and efficiency.
- Strategic Insights:
  - Optimal Sampling Strategy: Random Corrupt Negative Sampling emerges as the most effective strategy, optimizing the quality of embeddings.
  - Interplay of Techniques: The choice of negative sampling methods and embedding models plays a crucial role in refining KGQA frameworks, suggesting a targeted approach for future enhancements.