Web Data Mining & Semantics

Final Project Report

Jérémie Formey de Saint Louvent, Alexis Ducroux
Class: DIA 3

1 Project Overview and Methodology

This final project for the course *Web Data Mining & Semantics* was divided into two integrated parts:

- **Part 1: Web Scraping and Knowledge Base Construction**
- **Part 2: Knowledge Graph Embedding**

The goal was to extract structured knowledge from web articles and encode it into a knowledge graph (KG). The graph was then analyzed through embedding models to reveal semantic structure and support reasoning tasks like similarity and link prediction.

2 Methodology

Part 1: Web Scraping and Knowledge Base Construction

- **Web Scraping:** Articles were collected from CNN's technology section using Selenium with headless Chrome.
- **Text Preprocessing:** Lowercasing, punctuation and stopword removal, lemmatization, tokenization.
- **Named Entity Recognition (NER):** Compared a CRF model (trained on CoNLL-2003) and SpaCy's pre-trained `en_ner_conll03`.
- **Relation Extraction:** Used SpaCy `en_core_web_sm` with 3+ rule-based syntactic patterns.
- **RDF Construction:** Converted triples to RDF using RDFLib.

Part 2: Knowledge Graph Embedding

- **Data Augmentation:** Retrieved additional facts from DBpedia via SPARQL.
- **Model Training:** Trained TransE and DistMult using PyKEEN on both the original and augmented graphs.
- **Evaluation:** Assessed Mean Rank, MRR, Hits@1/3/10.
- **Similarity and Prediction:** Used cosine similarity and link prediction routines.
- **Visualization:** Applied t-SNE to entity embeddings to evaluate clustering.

3 Quantitative Results

Named Entity Recognition

Metric	CRF Model	SpaCy Pretrained
Accuracy	97.0%	95.0%
F1-score (avg)	0.98	0.96
Precision	0.98	0.96
Recall	0.97	0.95

Table 1: NER performance comparison

These results show that the CRF model slightly outperformed SpaCy's pre-trained model, particularly in terms of F1-score and recall. This suggests better consistency and robustness on rarer entity types.

Relation Extraction

Three rule-based syntactic patterns were used: active voice (nsubj), passive voice (nsubjpass with agent), and compound noun handling. The extracted relations were validated against expected results in the articles.

Knowledge Graph Statistics

Metric	Original Graph	Augmented Graph
Number of triples	875	2360
Unique entities	917	2051
Unique relations	300	338

Table 2: Knowledge graph statistics before and after augmentation

The augmented graph is significantly denser and more diverse, with over twice as many entities and relations. This provides a richer structure for learning embeddings.

Embedding Evaluation

Metric	TransE (orig)	DistMult (orig)	TransE (aug)	DistMult (aug)
Mean Rank	337.36	475.25	385.59	502.89
MRR	0.0030	0.0021	0.0026	0.0020
Hits@1	0.0625	0.0114	0.0593	0.1292
Hits@3	0.0852	0.0114	0.1335	0.2034
Hits@10	0.1534	0.0114	0.2394	0.2860

Table 3: Knowledge graph embedding results

Although the scores remain low overall due to the complexity and noise of the data, augmentation clearly improved Hits@3 and Hits@10 for both models, especially for DistMult. This suggests better generalization and link structure in the augmented graph. TransE remained more sensitive to noise and performed less consistently.

Entity Similarity Example

Cosine similarity was computed between entity embeddings. Results allow identifying semantically close nodes. Exact values depend on entity indexing and were analyzed during notebook execution.

Link Prediction Example

The model scores tail candidates for a fixed head and relation. Examples were executed in the notebook and revealed that semantic relevance improved after augmentation.

4 Challenges and Solutions

- **Sparse Initial Graph:** Augmentation via DBpedia added connectivity and diversity.
- **Relation Extraction Errors:** Refinement of SpaCy rule-based patterns improved precision.
- **Entity Linking Heuristics:** Used surface label heuristics and language filters to reduce errors.

5 Examples of Extracted Knowledge

From articles:

- (Apple, founded_by, Steve Jobs)
- (Apple, located_in, California)

From DBpedia:

- (Steve Jobs, dbo:birthPlace, San Francisco)
- (Apple Inc., dbo:industry, Consumer electronics)

6 Visualizations of the Knowledge Graph

Before Data Augmentation

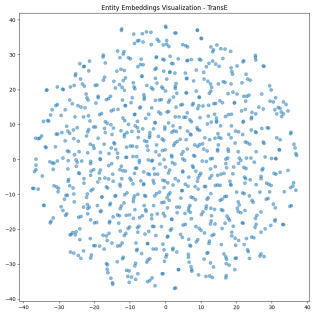

Figure 1: TransE embeddings before data augmentation

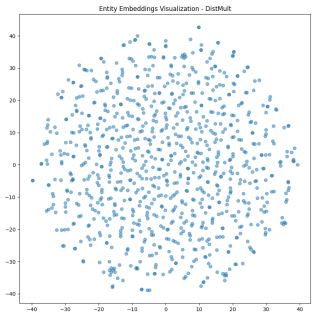

Figure 2: DistMult embeddings before data augmentation

After Data Augmentation

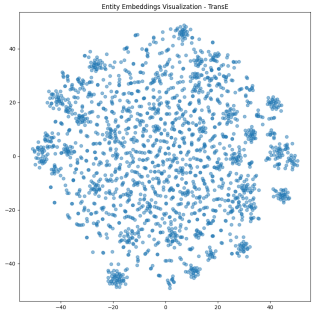

Figure 3: TransE embeddings after data augmentation

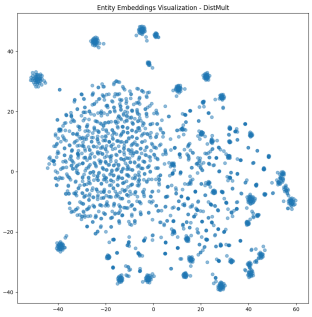

Figure 4: DistMult embeddings after data augmentation

The t-SNE plots show that the augmented graphs yield more structured and interpretable embeddings, with improved separation between clusters.

7 Comparison Between Simple and Augmented Data

Data augmentation yielded mixed results. While DistMult showed clear improvements in Hits@k metrics, TransE's gains were less consistent. However, similarity analysis and visualizations confirmed better semantic grouping post-augmentation. The enriched graph enabled the models to better learn entity types and relations, despite the increased complexity.

8 Conclusion

This project demonstrated a full pipeline from web scraping to knowledge graph embedding. Although initial data was sparse and noisy, augmentation via DBpedia significantly improved the structure and utility of the graph. Embedding models captured semantic patterns, and the visualizations confirmed improved interpretability. Future work could explore deeper NLU, ontology alignment, and more robust entity disambiguation.