Towards Zero-shot Knowledge Graph building: Automated Schema Inference

Salvatore Carta
salvatore@unica.it
Department of Mathematics and Computer Science, University of Cagliari
Cagliari, Italy

Alessandro Giuliani
alessandro.giuliani@unica.it
Department of Mathematics and Computer Science, University of Cagliari
Cagliari, Italy

Marco Manolo Manca
marcom.manca@unica.it
Department of Mathematics and Computer Science, University of Cagliari
Cagliari, Italy

Leonardo Piano
leonardo.piano@unica.it
Department of Mathematics and Computer Science, University of Cagliari
Cagliari, Italy

Sandro Gabriele Tiddia
sandrog.tiddia@unica.it
Department of Mathematics and Computer Science, University of Cagliari
Cagliari, Italy

ABSTRACT

In the current Digital Transformation scenario, Knowledge Graphs are essential for comprehending, representing, and exploiting complex information in a structured form. The main paradigm in automatically generating proper Knowledge Graphs relies on predefined schemas or ontologies. Such schemas are typically manually constructed, requiring an intensive human effort, and are often sensitive to information loss due to negligence, incomplete analysis, or human subjectivity or inclination. Limiting human bias and the resulting information loss in creating proper Knowledge Graphs is paramount, particularly for user modeling in various sectors, such as education or healthcare. To this end, we propose a novel approach to automatically generating a proper entity schema. The devised methodology combines the language understanding capabilities of LLM with classical machine learning methods such as clustering to properly build an entity schema from a set of documents. This solution eliminates the need for human intervention and fosters a more efficient and comprehensive knowledge representation. The assessment of our proposal concerns adopting a state-of-the-art entity extraction model (UniNER) to estimate the relevance of the extracted entities based on the generated schema. Results confirm the potential of our approach, as we observed a negligible difference between the topic similarity score obtained with the ground truth and with the automatically generated schema (less than 1% on average on three different datasets). Such an outcome confirms that the proposed approach may be valuable in automatically creating an entity schema from a set of documents.

UMAP Adjunct '24, July 01–04, 2024, Cagliari, Italy

ACM ISBN 979-8-4007-0466-6/24/07
https://doi.org/10.1145/3631700.3665234

CCS CONCEPTS

• **Information systems** → **Document topic models**; **Language models**; *Clustering and classification.*

KEYWORDS

Ontology Learning, Large Language Models, Named Entity Recognition

ACM Reference Format:
Salvatore Carta, Alessandro Giuliani, Marco Manolo Manca, Leonardo Piano, and Sandro Gabriele Tiddia. 2024. Towards Zero-shot Knowledge Graph building: Automated Schema Inference. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization (UMAP Adjunct '24), July 01–04, 2024, Cagliari, Italy.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3631700.3665234

1 INTRODUCTION

Nowadays, in the expanding scenario of Digital Transformation, Knowledge Graphs (KGs) represent crucial tools for understanding, organizing, and representing complex information in a structured form [5]. KGs can, on the one hand, facilitate data integration from heterogeneous sources and enhance the retrieval of information. On the other hand, they can enable advanced reasoning and inference capabilities [10]. However, the generation of KGs typically depends on the availability of pre-defined schemas or ontologies, which compose the backbone for defining entities and their relationships [3]. The schema entries describe the main structure of a KG, expressing entity types (typically organized in a hierarchical structure), their relationships, entity properties, constraints, and rules. Such schemas are typically manually built, requiring considerable human effort and domain expertise.

Relying on a manual building implies several challenges. First, inconsistencies and errors often affect manual building due to naturally occurring human mistakes, negligence, or incomplete analysis. Furthermore, as with most human-based tasks, the construction is prone to subjectivity or bias, as it relies strongly on the expertise and perspective of the human operators involved in the process. Moreover, due to the manual labor-intensive process, the generated schema cannot adapt to the continuous evolution of domains

and the rise of new information, leading to outdated knowledge representation or information loss. Such issues highlight the need for automated tools able to generate efficient schemas without any human support correctly.

This paper proposes an innovative approach to address the aforementioned challenges. The proposed methodology leverages the recent technological advancements of the well-known generative Large Language Models (LLMs).

Our proposal aims to reduce the need for extensive human effort to generate schemas while fostering a more efficient and exhaustive knowledge representation. In particular, our schema generation can reduce the human bias in building KGs, which are crucial for user modeling in various sectors, like education or healthcare. Let us remark that although LLMs are trained on data coming from humans, hence capturing the subjectivity points, they can mitigate the bias problem from several perspectives. First, LLMs can handle vast amounts of data by recognizing hidden patterns and relationships that humans cannot identify. Furthermore, such models are usually trained on numerous datasets covering a wide range of sources, exposing diverse points of view. Unlike classical methods, the proposed automated methodology can dynamically adapt to the characteristics of the input data, ensuring that the generated schema is not only tailored to the specific domain and context but can also be constantly updated to reflect the evolution of the underlying information.

For the sake of clarity, we specify that we only address entity schema generation. Nevertheless, we deem that the proposed method can also be easily adapted to find relations. Future studies will further investigate this aspect and generate complete schemas of entities and relationships.

To assess the efficacy of our entity schema generation method, we conducted several experiments on real-world datasets by performing a Named Entity Recognition (NER) task using the generated schemas. Specifically, we compared the quality and quantity of the entities extracted with the automatically generated schema against those extracted with the human-annotated schema. The experiments conducted show promising outcomes, indicating the potential of our approach to significantly support the process of KG construction.

The rest of the paper is organized as follows: 2 briefly describes the state-of-the-art; 3 describes our methodology, whereas 4 reports the experimental results. 5 ends the paper with the conclusions.

2 RELATED WORK

A Schema defines the structure and organization of the corresponding KG, establishing the set of entity types and relationships that compose the graph. Such a schema can be considered a simplified representation of an ontology, although entire ontologies are often used as a schema. Automatically extracting an ontology from text is a problem explored in literature over the last decade. Maedche et al. [9] proposed *Text-to-Onto*, a semi-automated system that builds a domain ontology from an initial core ontology using data mining and NLP. A refined version of Text-to-Onto has been released by Cimiano et al. [2], where *tf-idf* and *c-value/nc-value* methods are applied to find the relevance of terms. *HASTI* is one of the first automated Ontology Learning Frameworks [17]. It employs Lexico-syntactic patterns and semantic templates to extract concepts. Additionally, semantic templates, heuristic clustering analysis, and logical inference are used for both taxonomic and non-taxonomic relation extraction. With the recent advent of transformers, more advanced NLP techniques have been applied to the context of Ontology learning. Oba et al. [12] resolved the ontology generation task as a relationship classification between phrases. Their method consists of two steps, the first being to extract the key concepts and the second employing a BERT classifier to determine the relationship between each pair of concepts. Such relationships include *Synonym*, *Hyponym*, *Hypernym*, and *Unrelated*. This strategy permits the generalization of a taxonomic structure consisting of a hypernym–hyponym relationship from the phrase set. Similarly, the proposal of Oksanen et al. [13] leverages a BERT-based methodology to automatically extract ontologies from product reviews using a limited amount of hand-annotated training data. Saravanan et al. [14] focused on constructing an agricultural ontology. Their method involves extracting domain terms efficiently through term extraction methods and identifying relationships between entities using BERT with regular expressions and Graph Neural Networks (GNN). The work of Seo et al. [15] proposes an active learning framework for KG Schema Expansion. They implemented two neural methods for entity and relationship classification and exploited an active learning strategy to determine which types need expansion, considering their granularity. Most of the aforementioned methodologies are based on supervised methods. Due to their Natural Language understanding capabilities, Large Language Models (LLMs) have recently advanced the development of zero-shot Ontology learning methods. Funk et al. [6] introduced a method for building concept hierarchies in the form of directed acyclic graphs using generative LLMs like GPT 3.5. Their method needs a broad seed concept that will represent the context domain. They iteratively crawl the entire hierarchy from such a concept by querying the language model for relevant subconcepts. In such a context, *OntoChat* [20] is a framework that exploits a conversational agent to support ontology development and engineering. Moreover, ChatGPT can be prompted to explicitly generate an OWL ontology on sustainability, allowing the model to generate additional classes and properties [18].

3 METHODOLOGY

This Section presents the methodology devised to automatically infer the entity schema from a collection of documents. Figure 1 summarizes the proposed pipeline, which comprises four main modules: *Keyword Extraction*, *Topic Discovery*, *Topic Clustering*, and *Clusters Labeling*. We detail each module in the following.

3.1 Keywords Extraction

The proposed pipeline aims to generate an exhaustive entity schema. Therefore, finding and understanding which concepts are important and recurrent within the documents is essential. Given a set of Documents **D**, the role of the module is to identify a set of keywords **K** that will contain the relevant terms to represent the sets **D**. To this end, we leveraged the *KeyphraseTransformer*[1], a fine-tuned

[1] https://github.com/Shivanandroy/KeyPhraseTransformer

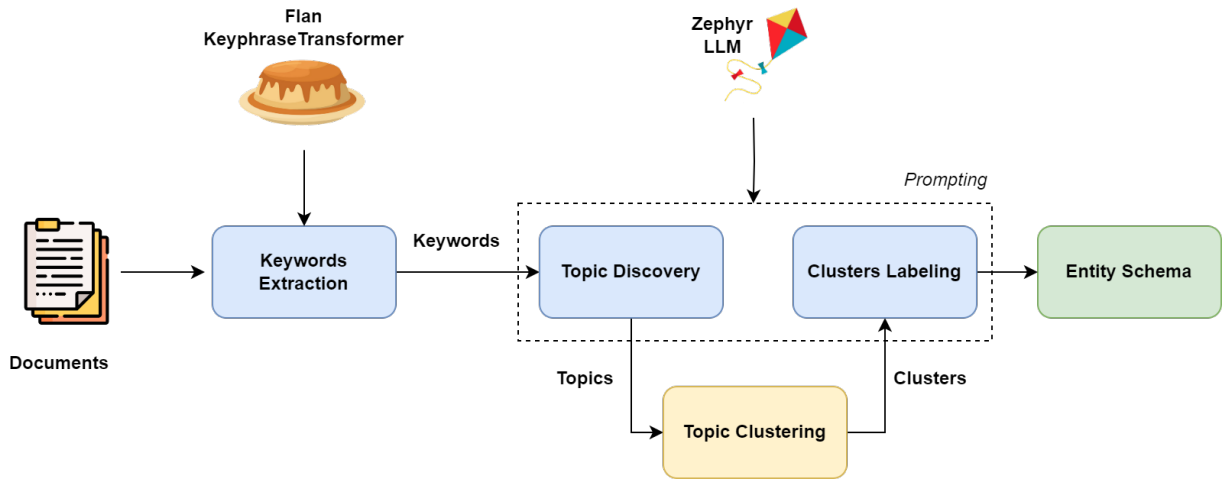

Figure 1: Automatic Entity Schema Generation pipeline: blue modules utilize generative LLMs, yellow relies on text embedding and unsupervised clustering, and green represents the output

version of the `FLAN-T5 LLM` [1] on the keyphrase extraction task. Therefore, for each document $d_i \in D$, we find a keyword set k_i by running the *KeyPhraseTransformer* on d_i. Finally, all the unique keywords belonging to the keywords sets $k_1, \ldots, k_n$ are merged in a final set $\mathbf{K}$.

3.2 Topic Discovery

A keyword set contains terms specific to each document, whereas an entity schema necessitates broader terms applicable across various contexts. Therefore, this module is aimed at transforming keywords into more generic topics. For this purpose, we exploit the generative Large Language Model `Zephyr` [19] with a prompt actively engineered to achieve the designated goal. The final version of the prompt is detailed below. Among the wide choice of LLMs currently available, we chose `Zephyr` as it is open and locally executable even with limited hardware. Indeed, as we observed in our preliminary experiments, it performs adequately well in information extraction tasks. However, we plan on testing even other and more recent LLMs. Accordingly, for each set of keywords k_i, we instruct `Zephyr` with the designed prompt, requesting a hypernym for each input keyword. The LLM will then output a JSON object formatted as `{<original keyword>: <closest hypernym>, ... }`. Finally, for each generated topic, we count its occurrences and remove topics that occur less than a threshold θ to filter out less meaningful topics (more details on the selected θ are given in Section 4.1).

Topic Assignment Prompt

```
Given a list of keywords, we identify their
closest hypernym and rewrite the input list by
adding the assigned hypernym. The output will be
 formatted as a JSON object with each keyword
and its corresponding hypernym.

Example:

Input:
- apple
- fruit
- food

Output:
{
  "apple": "fruit",
  "fruit": "food",
  "food": "edible item"
}
```

3.3 Topic Clustering

While the topic discovery module generalizes terms representing the keywords, such terms still turn out to be overly numerous. Above all, many semantically similar terms represent the same concept. Thus, the role of this module is to group semantically similar terms into separate sets (e.g., clusters). To this end, we first aggregate the keywords by their topic labels, obtaining, for each topic, a set of keywords whose hypernym is the topic label.

To perform an appropriate topic clustering, we need a set of features for each topic. Such a set is determined by relying on an adequately *distributed* text representation. In detail, given a topic, we project the label and all associated keywords in an embedding space (the selected embedding model is described in Section 4). To this end, a proper embedding strategy has been devised. The final topic embedding is obtained by a weighted average of the label embedding (with an associated weight w_t) and the centroid of the keyword embeddings (weight w_k). The aforementioned strategy has been empirically devised after a preliminary investigation. Such an exploratory stage gave rise to the following issues:

- considering an embedding of only the topic label is not the appropriate choice, as the `FLAN` model returns a correct hypernym, but often "too" generic. As an example, let us consider the keywords *apple*, *banana*, and *orange*. The topic discovery module returns the topic label *food*, which is correct but can be semantically far from the specific concept. For instance, the hypernym *fruit* would seem more appropriate and semantically similar.

- conversely, considering only the keywords showed the opposite issue, i.e., the inability to properly generalize, entailing the risk of grouping concepts that highly differ from a semantic perspective.

To this end, the proposed weighted strategy permits us to improve the topic representation by including and balancing the semantic information from both components (label and keywords). We deem that it expresses a proper trade-off for addressing the two major issues described above.

As a clustering technique, we considered *Hierarchical Clustering* (HCA) [11] as an appropriate algorithm for our problem, as the number of clusters does not have to be specified in advance. There are two main strategies to implement HCA: *Agglomerative* (bottom-up) or *Divisive* (top-down) strategies. We opted for the Agglomerative Clustering approach, where the algorithm starts by separating all data points in single clusters and iteratively merges the closest pairs, leveraging a similarity distance metric until one cluster only is left. In detail, we employ *cosine similarity* as distance metric and *complete-linkage* as agglomerative method[2], and an automatic search[3] of the *linkage distance threshold* that returns the best clusters according to the Silhouette score [16].

We further process the clusters, using the keyword counts introduced at the end of Section 3.2 to automatically filter out those not related to enough keywords mentioned in the original texts. For each cluster, we keep the topics that have a keyword count of at least n keywords. Subsequently, if the sum of the keyword count of each remaining cluster does not reach a given threshold m, we remove the cluster, as it is considered not significant enough.

3.4 Cluster Labeling

After generating clusters, to successfully compose the entity schema, assigning an adequately expressive and representative label for that group of terms is necessary. The final label acts as an entity type. To this end, we envision a new prompting step in which we query the LLM `Zephyr` to assign a label to each cluster. In particular, we have engineered the following prompt:

Cluster Labeling Prompt

```
Given a cluster of specific topics, your task is
 to identify a broader concept or category that
encompasses all the topics in the cluster.
This broader concept is referred to as an
ontology label. The label represents a general
category that includes all the specific topics.

For example, if the cluster includes topics such
 as "dog breeds", "dog training", and "dog
health", the ontology label could be "dog care".

The abstract description of the label should
have no individual examples, and would be: "The
process of caring for dogs.".

The output is a JSON document that includes the
ontology label and abstract description:
{ "ontology label": "dog care",
  "abstract description": "The process of caring
   for dogs."}
```

The output is the final entity schema, represented, as also reported in the previous prompt, with a proper JSON document.

4 EXPERIMENTS

This Section reports the experiments performed to assess our approach. We first present the experimental settings, giving more details about the implementation and the setting of the parameters mentioned in Section 3 to run the pipeline. We then discuss the datasets we selected and how we assessed the schema extracted from each corpus with a qualitative and quantitative evaluation.

4.1 Experimental Setup

The entire pipeline is implemented using the *Python* programming language[4], relying on the *transformer* package[5] for executing the LLM-related tasks, *sentence-transformer* for the text embeddings[6] required in the distributed representations of topics, and on *scikit-learn* for the implementation of the agglomerative clustering algorithm[7] and Silhouette score[8]. We run the pipeline by empirically setting each parameter mentioned in section 3. We used $w_t = 0.50$, $w_k = 0.50$ to obtain the distributed topics representation; $\theta = 1$, $n = 3$, $m = 9$ to keep a conservative number of clusters; $s = 0.05$ to search for the best distance threshold.

4.2 Datasets

To assess the automatic schema generation capabilities, we collected a set of datasets for named entity recognition, including a corpus and a ground truth schema. The need for distinct and specific domains drove our dataset selection process. Several well-known and classic NER datasets consist of large corpora from news, hence covering a wide selection of topics, but whose annotations cover only overly broad entity types, such as ORGANIZATION, PERSON, LOCATION, or MISC. Their texts would not be suitable for description by a simple schema, making our pipeline evaluation much more challenging. Accordingly, we selected three domain-specific datasets considered appropriate for testing.

- MIT-RESTAURANTS [8] contains 1520 development sentences related to food and restaurants;
- MIT-MOVIES [8] contains 2442 development sentences related to cinema and movies;
- BIO-NLP 2004 [4] contains 1927 development sentences related to molecular biology.

All selected dataset corpora are subdivided into the training, development, and test folds. We run our experiments only in the development corpora. Moreover, let us point out that the selected datasets are composed of single sentences. Since keyword extraction is usually done at the document level rather than at the sentence level, we group and concatenate sentences from each dataset to obtain texts of 20 sentences to feed to our pipeline.

[2]Searches for the maximum distance between topics of cluster pairs.
[3]Exhaustive search in $[0, 1]$ with step s, possible thanks to the limited range in which the cosine distance is defined.

[4]We plan to release a proper implementation of the proposed system.
[5]https://huggingface.co/docs/transformers/main_classes/pipelines
[6]https://huggingface.co/sentence-transformers/all-mpnet-base-v2
[7]https://scikit-learn.org/stable/modules/clustering.html#hierarchical-clustering
[8]https://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html

4.3 Evaluation

We evaluate the schemas generated by the devised pipeline applied to the datasets described in Section 4.2. A valid schema should correctly and comprehensively represent the content by providing a complete set of entity types coherently found in the underlying corpora. However, as there is no way of uniquely grouping and representing the topics encompassed by a corpus, the evaluation method should account for this. To this end, we relied on two different strategies for our evaluation:

- A *qualitative evaluation*, in which we manually compare the extracted schema with the dataset ground truth schema. We consider the actual semantics behind each label and map each type from one schema with one of the types from the other, giving us a valid reference to qualitatively interpret whether the automatically extracted schema is modeling the proper topics.
- A *quantitative evaluation*, in which we systematically evaluate the exhaustiveness of the schema based on the entities found in the text. We assess the alignment between the source text and the entities extracted from it by running a NER tool with a given schema and comparing the scores obtained by both the ground truth and automatic schema.

To extract the named entities required in different steps of our evaluation strategy, we rely on UniNER [21], an LLM-based named entity recognition tool capable of retrieving entities with custom types it receives as input.

4.3.1 ***Qualitative Evaluation.*** This Section illustrates the qualitative evaluation of our proposed entity schema generation method. Tables 1, 2, and 3 show the qualitative comparisons between the automatically extracted and ground truth types for each selected dataset. The left column reports the ground truth schema with the entity types identified by a human. The right column reports the extracted schema, with entity types automatically retrieved by our pipeline.

Operating from the left column, we manually aligned our schema so that each row has equivalent types/topics. When a type from one schema does not have an equivalent in the other, we use the *hash* symbol (#) to denote no match.

All Tables highlight how most of the ground truth types have a match in the automatically generated schema. Exceptions are a few types related to raw data, such as numeric values represented by the Ratings, Hours, and Price types found in Table 1, as these kinds of terms are not retrieved as keywords by the Keyphrase Transformer we adopted.

On the other hand, for all types related to actual entities, the automatically generated schema turns out to be more fine-grained. For example, in regards to the type Restaurant name in Table 1, our pipeline identified different types of establishments such as Pizza restaurant, Fast food establishment, and Coffee establishment. Again, the entity type Protein in Table 3 matches different protein-related concepts such as Protein kinase, Enzyme, Growth factor and more.

In the specific case of the BIO-NLP 2004 dataset (Table 3), our approach discovered several additional entity types not mentioned in the ground truth. Therefore, we aimed to investigate whether these brand-new types might be related or not to significant and valid entities. Table 4 reports a brief analysis including, for each new type, the number of unique entities founded by UniNER, along with a subset of actual entity mentions. Exploiting the automatically generated schema, UniNER finds a total of 269 unique entities distributed among seven new types not mentioned in the ground truth. We deem that automatic methods, such as the one presented in this work, represent useful tools to augment existing datasets or support human annotators in devising a proper entity schema for new ones.

4.3.2 ***Quantitative Evaluation.*** To quantitatively estimate the comprehensiveness of the automatically extracted schema, we employed an evaluation metric named *Topical Similarity Score* (TS), introduced by Jiang et al. [7], which was used to measure the information abundance of the extracted triples compared to the source documents. In our case, however, we use the same metric to quantify the information abundance of the entities extracted with UniNER according to a given schema. A comprehensive schema guides the NER tool in better retrieving the entities mentioned in the text, increasing the TS score, and indirectly measuring the schema quality and completeness. We compute the score for both the ground truth schema and the automatically generated one, using the same extraction tool to allow for an actual comparison. The score is calculated from the *KL-divergence* between the probability distribution of the latent topics identified in a document and the topics associated with the set of extracted entities. The probability distribution of the alignment of a text with the N latent topics is identified by a *Latent Dirichlet Allocation* model (LDA_N). For each document D, we define as E_D the set containing all entities extracted from D, encoded as "<entity label> (<type label>)", and concatenated all together. The Topical similarity is then calculated with the formula reported in Equation (1).

$$TS(D, E_D, N) = \exp\left(-\sum_{i=1}^{k} LDA_N(D)_i \cdot log\left(\frac{LDA_N(D)_i}{LDA_N(E_D)_i}\right)\right) \quad (1)$$

Tables 5, 6, and 7 report the comparison in terms of topical similarity scores between entities extracted according to the automatically generated schema (AG) and the ground truth schema (GT) when varying the number of latent topics (N) used for LDA topic modeling.

The reported results come from a single execution of the schema generation pipeline and NER tool. However, we performed several runs that returned barely different scores due to the probabilistic nature of the generative models involved. The higher the TS score is, the better the schema-related entities are aligned with the source text. Since the absolute value of the TS metric is not very informative, we focus on the relative differences between the scores of the entities extracted through both the AG and GT schema.

For all the datasets, we note that the automatically generated schema is complete and sufficiently exhaustive, as it achieves comparable performances to the ground truth with an average difference of less than a 1% on the TS score.

5 CONCLUSION

In the context of Digital Transformation, Knowledge Graphs (KGs) emerged as a revolutionary way of structuring and efficiently organizing complex information. In order to accurately and consistently represent information, KGs must pertain to an adequate schema. In addition, in automatically constructing a KG, it is necessary to pre-define the types of entities that comprise it. Establishing an exhaustive set of entity types that correctly semantically represent the set of documents can be challenging due to a lack of expertise or overly extensive document pools. Therefore, automatic methodologies to define or suggest a schema can be of great value in this regard. In this paper, we thus proposed an LLM-based automatic pipeline to automatically generate an exhaustive entity schema from a set of documents. In particular, after extracting relevant keyword sets, an LLM prompting strategy permits the inference of meaningful topics, which are

Ground Truth Types	Automatic Generated Types
Rating	#
Location	Urban neighborhood, Urban area
Amenity	Parking service
Cuisine	Mexican cuisine, Chinese cuisine
Hours	#
Price	#
Dish	Food
Restaurant Name	Pizza restaurant, Fast food establishment, Food establishment, Coffee establishment

Table 1: Comparison between ground truth entity types and inferred entity types on MIT-RESTAURANTS dataset

Ground Truth Types	Automatic Generated Types
Genre	Film genre
Year	#
Plot	#
Average ratings, Rating	Film rating
Actor	Film actor, Human, Female celebrity
Title	Movie, Science fiction movie, Fantasy film, Comedy
Song	#
Character	Fictional character
Review	#
Director	Film director
Trailer	#
#	Cinema, Horror cinema
#	Animated Medium

Table 2: Comparison between ground truth entity types and automatically inferred entity types on MIT-MOVIES dataset

Ground Truth Types	Automatic Generated Types
DNA	Gene regulation, Gene, Regulatory region, Promoter element
Protein	Protein, Protein complex, Protein kinase, Enzyme, Receptor, Growth Factor, Transcription Factor
Cell Type	Cell Biology, Cellular differentiation, Cell adhesion molecule, Cancer
Cell Line	Cell Line
RNA	Messenger RNA
#	Immunological activation, Antibody, Cytokine signaling
#	Chemical compound, Phorbol ester
#	Virus
#	Programmed cell death

Table 3: Comparison between ground truth entity types and automatically inferred entity types on BIO-NLP 2004 dataset

Entity Type	Unique Entities	Mentions sample
Immunological activation	29	*T-cell activation, immune responses, host defense response*
Antibody	25	*anti-CD4 mAb, ICAM-1, immunoglobulin*
Cytokine signaling	27	*IL-5 signaling, IL-6 signaling, cytokine gene transcription*
Chemical compound	134	*calcium, glucocorticoid, cortisol, flurbiprofen*
Phorbol ester	18	*TPA, phorbol 12-myristate 13-acetate, PMA*
Virus	31	*HIV-1, HIV-2, Epstein-Barr virus, human cytomegalovirus*
Programmed cell death	5	*apoptosis, apoptotic cell death, apoptotic process, gene knock-out*

Table 4: Qualitative analysis of the newly automatically discovered entity types not present in the ground truth of the BIO-NLP 2004 dataset

subsequently clustered and marked with a proper label. The experimentations demonstrate the potential of our approach in supporting the process of KG construction. For future work, we will focus on finding automatic ways of setting the parameter values we introduced in the pipeline, defining an

	N=5	N=10	N=20	N=30	N=40	N=50	N=75	N=100
AG	**0.939**	**0.850**	**0.630**	**0.519**	**0.583**	0.438	0.350	**0.421**
GT	0.930	0.807	0.630	0.514	0.520	**0.529**	**0.382**	0.343

Table 5: MIT-RESTAURANTS TS score of UniNER entities extracted with the automatic schema vs entities extracted with GT schema

	N=5	N=10	N=20	N=30	N=40	N=50	N=75	N=100
AG	0.908	0.802	0.771	0.680	0.670	0.616	0.557	0.524
GT	**0.929**	**0.845**	**0.822**	**0.753**	**0.705**	**0.699**	**0.635**	**0.571**

Table 6: MIT-MOVIES TS score of UniNER entities extracted with the automatic schema vs entities extracted with GT schema

	N=5	N=10	N=20	N=30	N=40	N=50	N=75	N=100
AG	**0.975**	**0.930**	**0.832**	**0.820**	**0.790**	**0.724**	**0.697**	**0.630**
GT	0.954	0.887	0.755	0.748	0.683	0.668	0.554	0.542

Table 7: BIO-NLP 2004 TS score of UniNER entities extracted with the automatic schema vs entities extracted with GT schema

iterative clustering strategy to generate a more complete and hierarchical schema, and including proper relationship types.

ACKNOWLEDGMENTS

We acknowledge financial support under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.5 - Call for tender No.3277 published on December 30, 2021 by the Italian Ministry of University and Research (MUR) funded by the European Union – NextGenerationEU. Project Code ECS0000038 – Project Title eINS Ecosystem of Innovation for Next Generation Sardinia – CUP F53C22000430001- Grant Assignment Decree No. 1056 adopted on June 23, 2022 by the Italian Ministry of University and Research (MUR). Also, Leonardo Piano, acknowledges financial support under the Ministerial Decree no. 351 of 9th April 2022, based on the NRRP – funded by the European Union - NextGenerationEU - Mission 4 "Education and Research", Component 1 "Enhancement of the offer of educational services: from nurseries to universities" - Investment 4.1, that provided financial support for his doctoral pathway.

REFERENCES

[1] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research* 25, 70 (2024), 1–53.

[2] Philipp Cimiano and Johanna Völker. 2005. Text2Onto. In *International Conference on Applications of Natural Language to Data Bases.* https://api.semanticscholar.org/CorpusID:263889270

[3] Kenneth Clarkson, Anna Lisa Gentile, Daniel Gruhl, Petar Ristoski, Joseph Terdiman, and Steve Welch. 2018. User-Centric Ontology Population. In *The Semantic Web*, Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam (Eds.). Springer International Publishing, Cham, 112–127.

[4] Nigel Collier and Jin-Dong Kim. 2004. Introduction to the Bio-entity Recognition Task at JNLPBA. In *NLPBA/BioNLP.* https://api.semanticscholar.org/CorpusID:7985741

[5] Lisa Ehrlinger and Wolfram Wöß. 2016. Towards a Definition of Knowledge Graphs.. In *SEMANTiCS (Posters, Demos, SuCCESS).*

[6] Maurice Funk, Simon Hosemann, Jean Christoph Jung, and Carsten Lutz. 2023. Towards Ontology Construction with Language Models. *ArXiv* abs/2309.09898 (2023). https://api.semanticscholar.org/CorpusID:262044094

[7] Pengcheng Jiang, Jiacheng Lin, Zifeng Wang, Jimeng Sun, and Jiawei Han. 2024. GenRES: Rethinking Evaluation for Generative Relation Extraction in the Era of Large Language Models. *arXiv preprint arXiv:2402.10744* (2024).

[8] Jingjing Liu, Panupong Pasupat, D. Scott Cyphers, and James R. Glass. 2013. Asgard: A portable architecture for multilingual dialogue systems. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (2013), 8386–8390. https://api.semanticscholar.org/CorpusID:14903208

[9] Alexander Maedche and Raphael Volz. 2001. The text-to-onto ontology extraction and maintenance system. https://api.semanticscholar.org/CorpusID:60483181

[10] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep Learning–based Text Classification: A Comprehensive Review. *ACM Comput. Surv.* 54, 3, Article 62 (apr 2021), 40 pages. https://doi.org/10.1145/3439726

[11] Fionn Murtagh and Pedro Contreras. 2012. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 2, 1 (2012), 86–97.

[12] Atsushi Oba, Incheon Paik, and Ayato Kuwana. 2021. Automatic Classification for Ontology Generation by Pretrained Language Model. In *Advances and Trends in Artificial Intelligence. Artificial Intelligence Practices: 34th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2021, Kuala Lumpur, Malaysia, July 26–29, 2021, Proceedings, Part I 34.* Springer, 210–221.

[13] Joel Oksanen, Oana Cocarascu, and Francesca Toni. 2021. Automatic Product Ontology Extraction from Textual Reviews. *arXiv preprint arXiv:2105.10966* (2021).

[14] Krithikha Sanju Saravanan and Velammal Bhagavathiappan. 2024. Innovative agricultural ontology construction using NLP methodologies and graph neural network. *Engineering Science and Technology, an International Journal* 52 (2024), 101675.

[15] Seungmin Seo, Byungkook Oh, Eunju Jo, Sanghak Lee, Dongho Lee, Kyong-Ho Lee, Donghoon Shin, and Yeonsoo Lee. 2022. Active Learning for Knowledge Graph Schema Expansion. *IEEE Transactions on Knowledge and Data Engineering* 34, 12 (2022), 5610–5620. https://doi.org/10.1109/TKDE.2021.3070317

[16] Ketan Rajshekhar Shahapure and Charles Nicholas. 2020. Cluster quality analysis using silhouette score. In *2020 IEEE 7th international conference on data science and advanced analytics (DSAA).* IEEE, 747–748.

[17] Mehrnoush Shamsfard and Ahmad Abdollahzadeh Barforoush. 2004. Learning ontologies from natural language texts. *International journal of human-computer studies* 60, 1 (2004), 17–63.

[18] Milena Trajanoska, Riste Stojanov, and Dimitar Trajanov. 2023. Enhancing knowledge graph construction using large language models. *arXiv preprint arXiv:2305.04676* (2023).

[19] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944* (2023).

[20] Bohui Zhang, Valentina Anita Carriero, Katrin Schreiberhuber, Stefani Tsaneva, Lucía Sánchez González, Jongmo Kim, and Jacopo de Berardinis. 2024. OntoChat: a Framework for Conversational Ontology Engineering using Language Models. *arXiv preprint arXiv:2403.05921* (2024).

[21] Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2023. UniversalNER: Targeted Distillation from Large Language Models for Open Named Entity Recognition. *ArXiv* abs/2308.03279 (2023). https://api.semanticscholar.org/CorpusID:260682557