Knowledge Graphs for Digital Transformation Monitoring in Social Media

Vanni Zavarella[1], Diego Reforgiato Recupero[1,] , Sergio Consoli[2], Gianni Fenu[1], Simone Angioni[1], Davide Buscaldi[3], Danilo Dessí[4] and Francesco Osborne[5,6]

[1]*Department of Mathematics and Computer Science, University of Cagliari, Cagliari, Italy*

[2]*European Commission, Joint Research Centre (DG JRC), Ispra (VA), Italy*

[3]*Laboratoire d'Informatique de Paris Nord, Sorbonne Paris Nord University, 99 Av. Jean Baptiste Clement, 93430 Villetaneuse, France*

[4]*Knowledge Technologies for Social Sciences Department, GESIS Leibniz Institute for the Social Sciences, Unter Sachsenhausen 6-8, Cologne, Germany*

[5]*Knowledge Media Institute, The Open University, Milton Keynes, UK*

[6]*Department of Business and Law, University of Milano Bicocca, Italy*

Abstract

Several techniques and workflows have emerged recently for automatically extracting knowledge graphs from documents like scientific articles and patents. However, adapting these approaches to integrate alternative text sources such as micro-blogging posts and news and to model open-domain entities and relationships commonly found in these sources is still challenging. This paper introduces an improved information extraction pipeline designed specifically for extracting a knowledge graph comprising open-domain entities from micro-blogging posts on social media platforms. Our pipeline utilizes dependency parsing and employs unsupervised classification of entity relations through hierarchical clustering over word embeddings. We present a case study involving the extraction of semantic triples from a tweet collection concerning digital transformation and show through two experimental evaluations on the same dataset that our system achieves precision rates exceeding 95% and surpasses similar pipelines by approximately 5% in terms of precision, while also generating a notably higher number of triples.

Keywords

Information Extraction, Knowledge Graphs, Social Media Analysis, Named Entity Recognition, Hierarchical Clustering, Word Embeddings

1. Introduction

In recent years, knowledge graphs (KGs) have become increasingly recognized for their ability to organize structured data in a semantically significant way, allowing them to effectively support various AI systems [1]. Large-scale KGs are usually generated through a semi-automated process,

3rd International Workshop on Knowledge Graph Generation from Text (TEXT2KG), co-located with ESWC 2024

Corresponding author.

v.zavarella@unica.it (V. Zavarella); diego.refogiato@unica.it (D. R. Recupero); sergio.consoli@ec.europa.eu (S. Consoli); gianni.fenu@unica.it (G. Fenu); simone.angioni@unica.it (S. Angioni); davide.buscaldi@lipn.univ-paris13.fr (D. Buscaldi); Danilo.Dessi@gesis.org (D. Dessí); francesco.osborne@gopen.ac.uk (F. Osborne)

utilizing both structured and unstructured data. Some prominent examples include DBpedia [2][1], Google Knowledge Graph[2], BabelNet[3], and YAGO[4]. A few proposals have been recently put forth for generating organized, interconnected, and machine-readable data frameworks of content found within text from microblogging platforms [3, 4, 5], using Semantic Web technologies such as ontologies and knowledge graphs [6, 7, 8]. However, creating extensive and high-quality knowledge graphs from social media is still an open research problem. Existing solutions either depend on systems that aid social media experts in structuring their knowledge, therefore suffering from scalability problems, or rely on information extraction pipelines [9, 10, 11]. Generating large-scale, coherent, and semantically sound representations of social media texts drawn from millions of posts has proven to be challenging, as existing methods for entity and relationship extraction typically focus on specific domains [4].

In this paper we present Triplétoile, an enhanced information extraction architecture designed to extract and merge instances of open-domain entities from social media text and to identify and generalize various relationships among these entities by using hierarchical clustering, word embeddings and dimensionality reduction techniques.

The designed architecture is scalable and introduces a novel approach for entity extraction that leverages the dependency tree of an input sentence and a list of patterns validated by experts. It incorporates a module for unifying and grounding entity instances using external resources such as DBpedia. We also provide a use case application of the proposed architecture to a set of around 100k tweets extracted from the X/Twitter platform[5] from 2022 and concerning the digital transformation domain and we released the resulting knowledge graph. Finally, we conducted an assessment of Triplétoile by comparing it to several alternative solutions using a benchmark dataset consisting of 500 triples and show that it outperforms them in terms of accuracy, while at the same time generating a relatively higher number of triples.

2. Related Work

Numerous scholarly articles delve into the methodologies for generating knowledge graphs across different domains and under various constraints [12, 7, 13, 14, 15, 16]. These knowledge graphs are often enhanced and refined using link prediction techniques [17, 18]. The extraction of knowledge graphs from web sources to answer questions related to social networks [3], such as Twitter or Facebook, has been widely discussed in literature [19, 20, 4]. He et al. [5] described how to build knowledge graphs for social networks by developing deep Natural Language Processing models. A number of information extraction pipelines have been proposed to create high-quality knowledge graphs within the social network analysis domain ([9, 4, 10, 11]).

Haslhofer et al. [21] have emphasized the importance of connected knowledge graphs and discovery, whereas Hyvönen and Rantala [22] have highlighted the significance of new relationships extracted from the original dataset. In recent years there has been also an increasing research focus on ontologies and interoperable data [23]. In particular, Dessì et al. [14] have

[1] https://www.dbpedia.org/
[2] https://developers.google.com/knowledge-graph
[3] https://babelnet.org/
[4] https://yago-knowledge.org/
[5] https://twitter.com/

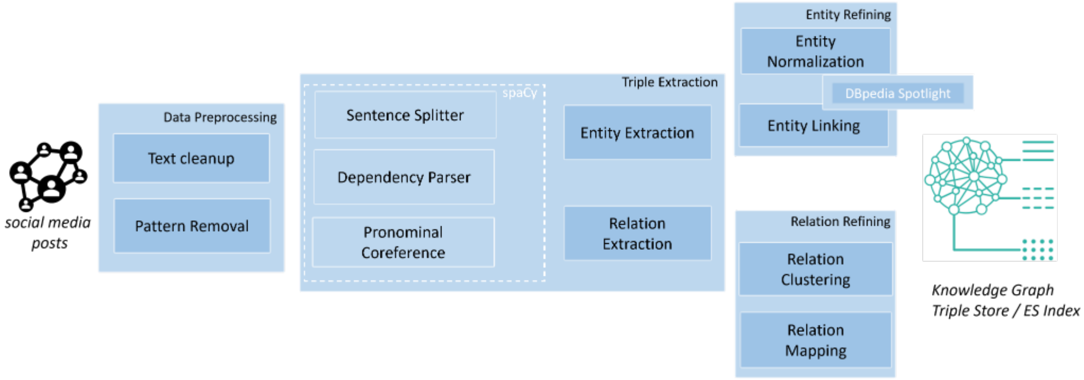

Figure 1: Flowchart of the pipeline for generating a knowledge graph from micro-blogging text data.

proposed an information extraction method that combines data from different tools using a domain ontology, enabling the creation of expansive knowledge graphs. This first approach has been a source of inspiration for further research in the field [24, 25, 26, 27, 28]. Recently, approaches leveraging fine-tuning of pre-trained large language model such as GPT-3 have been proven to be effective in performing joint named entity recognition and relation extraction for complex hierarchical information [29, 30]. Some recent solutions also augment large language model by using knowledge injection methods in order to improve their performance in specific domains [31].

3. Methodology

Figure 1 shows the workflow of the pipeline that we propose in this paper. We describe in more detail in the following the main component processing blocks and modules.

3.1. Data Preprocessing

Prior to extracting triples, we follow a two-fold approach to tweet normalization. On the one hand, we remove tokens and token sequences encoding platform-specific metadata or denoting communicative conventions that (typically) do not carry any syntactic function in the tweet sentence, namely sentiment emoticons and smileys, reserved tokens (e.g., RT for ‘retweet’) and URLs. On the other hand, we keep by default other platform-specific tokens that can carry syntactic functions in some contexts, like the tokens (e.g. #digitaltransformation, #SME, @NASA). Then, we apply a number of heuristics for capturing and removing token patterns that typically disrupt the syntactic parsing of the sentence[6]. This results in fixing noisy parser edges induced for example by trailing hashtags sequences. The preprocessing step is carried out using the output of Spacy’s English transformer pipeline *en_core_web_trf-3.6.1* after customizing the default Tokenizer in order to parse tweet metadata (e.g., mentions and hashtags)[7].

[6] For example, for any sequence of size $n > 1$ hashtags/mentions/URL, we drop the sub-sequence with indexes $[1 : n]$ or drop the entire sequence if preceded by a sentence closing marker like (’ ’,’:’,’?’,’.’).
[7] https://github.com/explosion/spacy-models/releases/tag/en_core_web_trf-3.6.1

3.2. Triple Extraction

In the triple extraction block, preprocessed tweets are sentence split and each sentence is fed to the Spacy pipeline. Building upon the works in [32] and [33], we define a set of procedures to extract candidate nominal entities and predicative triples connecting them out of dependency parse trees.

Entity extraction module: It detects local nominal phrases with a restricted range of syntactic modifications (e.g., compound nouns and adjectives). It then connects and expands them with a. non-recursive attached prepositional phrases; b. quantity-type entities (MONEY, PERCENT, QUANTITY, CARDINAL); c. entity spans linked via pronominal anaphoras, resolved using the Spacy pipeline component coreferee[8]. Overall, the module ends up with a set $E = \{e\ , ..., e_n\}$ of non-unified, candidate entity phrases (e.g. *digitaltransformation* and *digital transformation* are not mapped to the same general concept DIGITAL TRANSFORMATION at this stage.

Relation extraction module: For each sentence s_i all the shortest paths of the dependency tree between each pair of entities (e_m, e_n) containing a verb and matching any of a shortlist of expert validated patterns[9] are selected. The target pattern set has been selected through an expert validation process over a sampling of the most frequent patterns matched in an external, open-domain text corpus. The entire updated process generates a set of verbal relations $V = v\ , ..., v_k$ and a set of triples $S = s\ , ..., s_k$ of the form $< e_m, v, e_n >$ where $v \in V$ and $e \in E$.

The final goal of the pipeline is to allow to generalize from the set S of surface form triples to the lower sized set $T = t\ , ..., t_h$ of triples of the form $< \ _m, r, \ _n >$ where each $\ _i \in E$ is a unified entity and r is a label in a generalized relation vocabulary R.

3.3. Entity Refining

Entities are first cleaned up by removing leading/trailing punctuation marks as well as stopwords. Then, hashtags and @ mentions are normalized and lower-cased and "camel case" forms resolved (e.g. from *SmartCities* to *'smart cities'*), while we lemmatize and lowercase all other component tokens whose POS tag is neither Verb nor Proper Noun.

We leverage the linking of these normalized candidate entities to DBpedia entries via DBpedia Spotlight library[10] in order to merge them. To this purpose, we run the library over modified tweet sentences with the original subjects and objects entity spans replaced with their normalized forms, and then merge entities that get linked to the same DBpedia entries. For example, the two candidate entities *'Gartner'* and *'@Gartner_inc'* are merged as they get linked to the DBpedia entry of the Gartner consulting firm http://dbpedia.org/resource/Gartner). This is then formalized with a relation `owl:sameAs` in the final knowledge graph. In case only the first condition is met, we assign a semantic 'relatedness' link between the candidate entity and the DBpedia entry, indicating that the former is not an instance of, but rather related to

[8] https://github.com/richardpaulhudson/coreferee

[9] https://github.com/zavavan/dtm_kg/blob/master/resources/paths.txt

[10] https://spacy.io/universe/project/spacy-dbpedia-spotlight

the latter. For example, the span *'@gartner_survey'* is considered only 'related' (encoded as `skos:related`) to the DBpedia entry for Gartner.

3.4. Relation Refining

In order to find the best predicate label r for each relation verb v in a triple $< e_m, v, e_n >$ and to map v to r in the resulting triple, we first derive a word embedding representation of the verb predicates from a pre-trained model, then we compute an optimized clustering of the relation vectors, and finally use a representative instance of each cluster to map verb predicates.

Relation Embeddings: For each single or multi-token relation predicate, we use the static, 300-dimensional word embeddings learned with GloVe [34] and made available for text Span objects in the Spacy *en_core_web_lg-3.6.0* pipeline[11][12].

Dimensionality Reduction and Clustering: We used the HDBSCAN clustering algorithm enhanced by previously applying UMAP dimension reduction technique on the word embeddings vectors[13]. HDBSCAN is a hierarchical version of the popular density-based DBSCAN algorithm, which is characterized by considering outliers and leaves unclustered the data points lying in low-density regions [35]. Consequently, high dimensional data require more observed samples to produce the suitable level of density for HDBSCAN to work properly. However, applying UMAP to perform non-linear, manifold aware dimension reduction [36] has been proven to transform the datasets down to a dimension small enough for HDBSCAN to cluster the vast majority of instances. In order to optimize the combination of UMAP and HDBSCAN, we perform a grid search over the hyper-parameters of both algorithms and evaluate the clustering using the score: $S = silhouette_X \cdot clustered_X$, where $silhouette_X$ is the mean silhouette coefficient over all the instances of the dataset X that were actually clustered by HDBSCAN [37] and $clustered_X$ is the fraction of instances of X that were actually clustered. In practice, we optimize for the classical measure of cluster cohesion and separation while penalizing the configurations with low coverage of the dataset. We finally chose a subset of best-scoring hyper-parameter configurations and plotted their S score over the number of output clusters they generate, so that we are able to pick a sub-optimal configuration that balances between generalization (fewer clusters) and accuracy (cluster number closer to the dataset size).

Relation Mapping: Finally, for each relation verb v in the dataset, we replace it with the predicate label r consisting of the lemma of the most frequent relation in the cluster of v. Otherwise, we map it to itself if v was an outlier.

[11] https://github.com/explosion/spacy-models/releases/tag/en_core_web_lg-3.6.0

[12] We tested using various contextual embeddings however it turned out that these representations were not suitable for generalizing enough over relations, probably due to the context-specific information they are encoding.

[13] https://umap-learn.readthedocs.io/en/latest/parameters.html

4. Evaluation

We first evaluate the precision by manually assessing the truthfulness of a test set of statements. Second, we evaluate our pipeline's precision against a number of alternative tools.

Human Expert Assessment: We randomly selected about 500 statements extracted by our pipeline and ask three domain expert evaluators to assess each statement as True or False[14]. Average pair-wise Cohen κ inter-rater agreement was 0.61, while Fleiss κ agreement score over all 3 raters (ranging in $[\ \ 1, +1]$, [38]) reached 0.558 (substantial agreement). The accuracy of the majority vote assessments over the 500 triples was 0.96, indicating that the pipeline is able to extract triples with good precision.

Comparative Evaluation: Successively, we randomly sampled 500 tweets from the 100k-sized original dataset and used our pipeline to extract candidate entities. We then merge this set of entities with the ones generated by the DyGIEpp Extractor [39]. Finally we deploy four alternative methods to identify relationships between these entities and extract statements from the 500 tweets. Specifically, we compared with: a. **OpenIE Extractor**, the IE tool of the Stanford Core NLP suite [40]; b. **PoST Extractor**, a module that searches for all verbs in a 15 token window between two candidate entities in a sentence to extract verb relations; c. **Dependency based Extractor**, a module that exploits 12 hand-crafted paths[15] over Stanford Core NLP dependency parses to find verbs that connect DyGIEpp entities; d. **Entity and Relationship Refiner**, described in [33].

Extraction Method	Generated Triples	Precision
OpenIE Extractor	588	0.52
PoST Extractor	**1 015**	0.17
Dependency-based Extractor	339	0.77
Entity and Relationship Refiner	348	0.31
Triplétoile	663	**0.82**

Table 1
Precision P) of the triples extracted from a set of alternative methods from a set of 500 tweets, using a combination of Triplétoile and DyGIEpp candidate entities.

Table 1 reports the number of extracted triples for each of these methods. In order to use these numbers as an indirect assessment of the relative recall levels of the different pipelines, we also report the expert-assessed precision on a limited random sample of 150 triples generated by each method. It can be seen how the precision of our pipeline on this smaller sample largely outperforms all the alternative methods, while also yielding the second largest number of triples, interestingly outperforming the Dependency-based Extractor method, which deploys very similar syntactic information from the sentence[16].

[14] For example, a triple like $< 78\%\_of\_\ \ healthcare\ USE\ Digital\_Transformation >$ would be marked as False if extracted from the text *'78% of healthcare organisations deploy DigitalTransformation'* as the head of the subject noun phrase of the relation is actually 'organisations'.

[15] https://github.com/danilo-dessi/SKG-pipeline/blob/main/resources/path.txt

[16] This may be due to the application of the processing step upstream of the triple extraction process.

Subject Entity	Relation	Object Entity
pandemic	accelerate	digital_transformation
artificial_intelligence	impact	insurance_sector
microsoft	buy	riskiq
data-driven_insight	drive	decision-making
hootsuite	buy	ai_chatbot_firm
automl	generate	data-driven_insight
image_classification	use	transfer_learning
image_recognition_framework	use	artificial_intelligence
hsbc_qatar	introduce	mobile_payment
ford_motor_company	explore	blockchain_technology

Table 2
A sample of statements extracted by the Triplétoile pipeline.

5. Digital Transformation Monitoring Knowledge Graph

The presented prototype pipeline was deployed as part of a Digital Transformation monitoring system, targeting specifically its capacity to link and extend existing knowledge graphs generated from conventional sources (scientific papers, patents) with continuous updates from news and social media. Therefore, we have generated a knowledge graph from around 100k topic-specific tweets, sampled from 4M English language tweets from 2022 containing the hashtag #DigitalTransformation, excluding retweets[17].

The generated DTSMM (Digital Transformation Social Media Monitor) knowledge graph comprises approximately 22,270 (non-reified) triples, connecting a total of 22597 nodes via 43428 edges. A number of sample triples are shown in Table 2.

We reified then each claim into *dtsmm-ont:Statement* class instances, where *dtsmm-ont:Statement* defines a specific claim extracted from a given number of tweets. Figure 2 shows an example of claim reification having the instance *multi_page_document_classification* as *rdf:subject.* DTSMM features a total of 18693 unique detected entities, whose 33.9 and 6.44 included hashtags and @ entity mentions, respectively, 3.34 were complex noun phrases with prepositional attachments, while around 16.6 contained quantitative modifiers of any type (currency, percent, etc.). Out of all the generated triples, a 5.98 had either the subject or object entity made by a resolved pronominal anaphora.

Around 8% of all unique entities were linked to DBpedia entries via 2,857 *owl:sameAs* and 3,309 *skos:related* predicates in order to encode entity equality and relatedness, respectively. Table 3 lists the 10 most frequent DBpedia entities linked by DTSMM. The most frequent DBpedia-inherited types in the graph are: DBpedia:Company (441 unique entities), DBpedia:Person (118), DBpedia:Website (92), DBpedia:Software (59), DBpedia:Bank (31), DBpedia:Politician (29) and DBpedia:City (29).

The primary use case of DTSMM fits within the research initiatives at the European Commission's Competence Center on Composite Indicators and Scoreboards[18] within the Joint Research

[17] We used the Twitter public API v2 full-archive search endpoint. Near-duplicate tweets were also removed.

[18] European Commission's Competence Center on Composite Indicators and Scoreboards (COIN): https://composite-indicators.jrc.ec.europa.eu/

Top Linked Entities
Artificial_intelligence
Pandemic
Digital_transformation
Coronavirus_disease_2019
Microsoft
Cloud_computing
Google
Salesforce.com
Gartner
Chatbot

Table 3
10 most frequent DBpedia-linked entities in the DTSMM knowledge graph.

Centre (JRC)[19], whose goal is to create a tracker that monitors societal and economic activities through European countries using unconventional data [41].

Therefore, DTSMM has been made publicly accessible both through a SPARQL endpoint and a serialization file. Through the Virtuoso SPARQL endpoint https://api-vast.jrc.service.ec.europa.eu/sparql/ DTSMM can be queried using the graph name 'DTSMM_KG'[20] as shown in Figure 3, where we retrieve all the statements in DTSMM having the target entity `dtsmm:microsoft` as object.

Finally, a Turtle format serialization of DTSMM has been publicly released[21] within the Joint Research Centre Data Catalogue[22], as well as within the European Data portal[23]. The direct link is: https://jeodpp.jrc.ec.europa.eu/ftp/jrc-opendata/CC-COIN/se-tracker/DTSMM_KG.ttl.

[19] The Joint Research Centre (JRC) of the European Commission (EC): https://ec.europa.eu/info/departments/joint-research-centre_en

[20] Currently the access is password protected, with credentials available upon request to authors.

[21] Under Creative Commons Attribution 4.0 International (CC BY 4.0)

[22] https://data.jrc.ec.europa.eu/dataset/f7be47f7-49a2-44e8-9dc8-043735af4139

[23] https://data.europa.eu/88u/dataset/f7be47f7-49a2-44e8-9dc8-043735af4139

```
dtsmm-ont:statement_10100 a dtsmm-ont:Statement,
 rdf:Statement ;
dtsmm-ont:negation false ;
dtsmm-ont:comesfromTweet dtsmm:tweet_1424266328882429952 ;
...
dtsmm-ont:hasSupport 6 ;
rdf:subject dtsmm:multi_page_document_classification ;
rdf:predicate dtsmm-ont:use ;
rdf:object dtsmm:machine_learning .
```

Figure 2: A shortened sample reification for a statement concerning the ontology instances *multi_page_document_classification* and *machine_learning*, with the data property *dtsmm ont:hasSupport* reporting the number of tweets grounding the claim.

```
PREFIX dtsmm: <http://dtsmmkg.org/dtsmmkg/resource/>
PREFIX dtsmm-ont: <http://dtsmmkg.org/dtsmmkg/ontology >
SELECT ?statement
FROM <DTSMM_KG>
WHERE { ?statement a rdf:Statement .
 ?statement rdf:subject dtsmm:microsoft . }
```

Figure 3: Query returning all DTSMM statements with the graph entity `dtsmm:microsoft` as `rdf:subject`.

6. Conclusions

We presented an approach to specifically extract a knowledge graph comprising open-domain entities from micro-blogging posts on social media platforms. In a topic-specific test collection of Digital Transformation tweets the pipeline proved to outperform some of the state-of-the-art methods, generating mostly valid triples. Moreover, around 12% of entity mentions are linked to DBpedia entries, suggesting that the method is potentially useful for tracking relevant entities in the target social media text collection.

A current limitation is that the entity and relation extraction processes are not backed by an underlying ontology specification. Therefore, on one hand, the extracted entities are not natively typed and no domain-specific classification schema for relations is available for setting up a supervised learning of relation mapping. We plan to work on an enhanced version of the pipeline that builds upon the current entity and relation spans and further classifies them into domain-specific categories, leveraging fine-tuning of contextual word embedding representations from Large Language Models [42]. Simultaneously, we aim to capitalize on the resultant knowledge graph to develop knowledge plugins [43], thus augmenting the proficiency of these language models across various natural language processing tasks.

Acknowledgements

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).

References

[1] C. Peng, F. Xia, M. Naseriparsa, F. Osborne, Knowledge graphs: opportunities and challenges, Artificial Intelligence Review (2023) 1–32.

[2] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, C. Bizer, DBpedia–A large-scale, multilingual knowledge base extracted from Wikipedia, Semantic Web 6 (2015) 167–195. doi:10.3233/SW-140134.

[3] P. S. Raji, S. Surendran, RDF approach on social network analysis, in: 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS), 2016, pp. 1–4. doi:10.1109/RAINS.2016.7764416.

[4] J. Dörpinghaus, S. Klante, M. Christian, C. Meigen, C. Düing, From social networks to knowledge graphs: A plea for interdisciplinary approaches, Social Sciences & Humanities Open 6 (2022) 100337. doi:10.1016/j.ssaho.2022.100337.

[5] Q. He, J. Yang, B. Shi, Constructing knowledge graph for social networks in a deep and holistic way, in: Companion Proceedings of the Web Conference 2020, WWW '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 307–308. doi:10.1145/3366424.3383112.

[6] G. Xiao, D. Calvanese, R. Kontchakov, D. Lembo, A. Poggi, R. Rosati, M. Zakharyaschev, Ontology-based data access: A survey, in: IJCAI International Joint Conference on Artificial Intelligence, volume 2018-July, 2018, p. 5511 – 5519. doi:10.24963/ijcai.2018/777.

[7] A. Hogan, The Semantic Web: Two decades on, Semantic Web 11 (2020) 169–185. doi:10.3233/SW-190387.

[8] L. Ehrlinger, W. Wöß, Towards a definition of knowledge graphs, in: M. Martin, M. Cuquet, E. Folmer (Eds.), Joint Proceedings of the Posters and Demos Track of the 12th International Conference on Semantic Systems - SEMANTiCS2016 and the 1st International Workshop on Semantic Change & Evolving Semantics (SuCCESS'16) co-located with the 12th International Conference on Semantic Systems (SEMANTiCS 2016), Leipzig, Germany, September 12-15, 2016, volume 1695 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2016, pp. 1–4. URL: https://ceur-ws.org/Vol-1695/paper4.pdf.

[9] C. Barbosa, L. Félix, V. Vieira, C. Xavier, Sara - A Semi-Automatic Framework for Social Network Analysis, in: Anais Estendidos do XXV Simpósio Brasileiro de Sistemas Multimídia e Web, SBC, Porto Alegre, RS, Brasil, 2019, pp. 59–62. doi:10.5753/webmedia_estendido.2019.8137.

[10] H. Alani, A. Gangemi, V. Presutti, D. Reforgiato Recupero, A. G. Nuzzolese, F. Draicchio, M. Mongiovì, Semantic Web Machine Reading with FRED, Semantic Web 8 (2017) 873–893. doi:10.3233/SW-160240.

[11] J. L. Martinez-Rodriguez, I. Lopez-Arevalo, A. B. Rios-Alvarado, OpenIE-based approach for Knowledge Graph construction from text, Expert Systems with Applications 113 (2018) 339–355. doi:10.1016/j.eswa.2018.07.017.

[12] P. Ristoski, H. Paulheim, Semantic web in data mining and knowledge discovery: A comprehensive survey, Journal of Web Semantics 36 (2016) 1 – 22. doi:10.1016/j.websem.2016.01.001.

[13] T. Tudorache, Ontology engineering: Current state, challenges, and future directions, Semantic Web 11 (2020) 125 – 138. doi:10.3233/SW-190382.

[14] D. Dessì, F. Osborne, D. Reforgiato Recupero, D. Buscaldi, E. Motta, Generating knowledge graphs by employing natural language processing and machine learning techniques within the scholarly domain, Future Generation Computer Systems 116 (2021) 253–264. doi:10.

1016/j.future.2020.10.026.

[15] D. Dessì, F. Osborne, D. Reforgiato Recupero, D. Buscaldi, E. Motta, H. Sack, AI-KG: An Automatically Generated Knowledge Graph of Artificial Intelligence, Lecture Notes in Computer Science 12507 (2020) 127 – 143. doi:10.1007/978-3-030-62466-8_9.

[16] S. Angioni, A. Salatino, F. Osborne, D. R. Recupero, E. Motta, Aida: A knowledge graph about research dynamics in academia and industry, Quantitative Science Studies 2 (2021) 1356–1398.

[17] A. Kumar, S. S. Singh, K. Singh, B. Biswas, Link prediction techniques, applications, and performance: A survey, Physica A: Statistical Mechanics and its Applications 553 (2020) 124289.

[18] M. Nayyeri, G. M. Cil, S. Vahdati, F. Osborne, M. Rahman, S. Angioni, A. Salatino, D. R. Recupero, N. Vassilyeva, E. Motta, et al., Trans4e: Link prediction on scholarly knowledge graphs, Neurocomputing 461 (2021) 530–542. doi:10.1016/j.neucom.2021.02.100.

[19] D. Collarana, M. Galkin, C. Lange, S. Scerri, S. Auer, M.-E. Vidal, Synthesizing knowledge graphs from web sources with the minte+ framework, in: Lecture Notes in Computer Science, volume 11137, 2018, p. 359 – 375. doi:10.1007/978-3-030-00668-6_22.

[20] E. Gabrilovich, N. Usunier, Constructing and mining web-scale knowledge graphs, in: SIGIR 2016 - Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2016, p. 1195 – 1197. doi:10.1145/2911451.2914807.

[21] B. Haslhofer, A. Isaac, R. Simon, Knowledge graphs in the libraries and digital humanities domain, Springer International Publishing, Cham, 2018, pp. 1–8. doi:10.1007/978-3-319-63962-8_291-1.

[22] E. Hyvönen, H. Rantala, Knowledge-based relation discovery in cultural heritage knowledge graphs, in: CEUR Workshop Proceedings, volume 2364, 2019, p. 230 – 239.

[23] S. Cristofaro, E. M. Sanfilippo, P. Sichera, D. Spampinato, Towards the representation of claims in ontologies for the digital humanities, in: CEUR Workshop Proceedings, volume 2949, 2021, pp. 1 –12. URL: https://ceur-ws.org/Vol-2949/paper6.pdf.

[24] Y. Xiao, C. Li, M. Thürer, A patent recommendation method based on kg representation learning, Engineering Applications of Artificial Intelligence 126 (2023). doi:10.1016/j.engappai.2023.106722.

[25] T. Man, A. Vodyaho, D. Ignatov, I. Kulikov, N. Zhukova, Synthesis of multilevel knowledge graphs: Methods and technologies for dynamic networks, Engineering Applications of Artificial Intelligence 123 (2023). doi:10.1016/j.engappai.2023.106244.

[26] S. Yu, C. Peng, C. Xu, C. Zhang, F. Xia, Web of conferences: A conference knowledge graph, in: WSDM 2023 - Proceedings of the 16th ACM International Conference on Web Search and Data Mining, 2023, pp. 1172–1175. doi:10.1145/3539597.3573024.

[27] G. Tamašauskaite, P. Groth, Defining a knowledge graph development process through a systematic review, ACM Transactions on Software Engineering and Methodology 32 (2023). doi:10.1145/3522586.

[28] A. Chessa, G. Fenu, E. Motta, F. Osborne, D. Reforgiato Recupero, A. Salatino, L. Secchi, Data-driven methodology for knowledge graph generation within the tourism domain, IEEE Access 11 (2023) 67567–67599. doi:10.1109/ACCESS.2023.3292153.

[29] A. Dunn, J. Dagdelen, N. Walker, S. Lee, A. S. Rosen, G. Ceder, K. Persson, A. Jain, Structured

information extraction from complex scientific text with fine-tuned large language models, 2022. arXiv:2212.05238.

[30] H. Khorashadizadeh, N. Mihindukulasooriya, S. Tiwari, J. Groppe, S. Groppe, Exploring in-context learning capabilities of foundation models for generating knowledge graphs from text., in: TEXT2KG/BiKE@ESWC, 2023, pp. 132–153. DBLP License: DBLP's bibliographic metadata records provided through http://dblp.org/ are distributed under a Creative Commons CC0 1.0 Universal Public Domain Dedication. Although the bibliographic metadata records are provided consistent with CC0 1.0 Dedication, the content described by the metadata records is not. Content may be subject to copyright, rights of privacy, rights of publicity and other restrictions.

[31] A. Cadeddu, A. Chessa, V. De Leo, G. Fenu, E. Motta, F. Osborne, D. R. Recupero, A. Salatino, L. Secchi, A comparative analysis of knowledge injection strategies for large language models in the scholarly domain, Engineering Applications of Artificial Intelligence 133 (2024) 108166.

[32] D. Dessì, F. Osborne, D. R. Recupero, D. Buscaldi, E. Motta, CS-KG: A large-scale knowledge graph of research entities and claims in computer science, in: U. Sattler, A. Hogan, C. M. Keet, V. Presutti, J. P. A. Almeida, H. Takeda, P. Monnin, G. Pirrò, C. d'Amato (Eds.), The Semantic Web - ISWC 2022 - 21st International Semantic Web Conference, Virtual Event, October 23-27, 2022, Proceedings, volume 13489 of *Lecture Notes in Computer Science*, Springer, 2022, pp. 678–696. doi:10.1007/978-3-031-19433-7_39.

[33] D. Dessì, F. Osborne, D. R. Recupero, D. Buscaldi, E. Motta, SCICERO: A deep learning and NLP approach for generating scientific knowledge graphs in the computer science domain, Knowledge-Based Systems 258 (2022) 109945. doi:10.1016/j.knosys.2022.109945.

[34] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1532–1543. URL: http://www.aclweb.org/anthology/D14-1162.

[35] C. Malzer, M. Baum, A hybrid approach to hierarchical density-based cluster selection, in: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, volume 2020-September, 2020, p. 223 – 228. doi:10.1109/MFI49285.2020.9235263.

[36] L. McInnes, J. Healy, J. Melville, Umap: Uniform manifold approximation and projection for dimension reduction, 2020. arXiv:1802.03426.

[37] F. Batool, C. Hennig, Clustering with the Average Silhouette Width, Computational Statistics and Data Analysis 158 (2021). doi:10.1016/j.csda.2021.107190.

[38] R. Falotico, P. Quatto, Fleiss' kappa statistic without paradoxes, Quality and Quantity 49 (2015) 463 – 470. doi:10.1007/s11135-014-0003-1.

[39] D. Wadden, U. Wennberg, Y. Luan, H. Hajishirzi, Entity, relation, and event extraction with contextualized span representations, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 5784–5789. doi:10.18653/v1/D19-1585.

[40] G. Angeli, M. J. Johnson Premkumar, C. D. Manning, Leveraging linguistic structure for open domain information extraction, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on

Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 344–354. doi:10.3115/v1/P15-1034.

[41] M. Colagrossi, S. Consoli, F. Panella, L. Barbaglia, Tracking socio-economic activities in european countries with unconventional data, in: ACM International Conference Proceeding Series, 2022, p. 323 – 330. doi:10.1145/3524458.3547242.

[42] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, X. Wu, Unifying large language models and knowledge graphs: A roadmap, 2023. arXiv:2306.08302.

[43] A. Meloni, S. Angioni, A. Salatino, F. Osborne, D. R. Recupero, E. Motta, Integrating conversational agents and knowledge graphs within the scholarly domain, IEEE Access 11 (2023) 22468–22489.