Ontology-Driven Extraction of Contextualized Information from Research Publications

Vayianos Pertsas[a] and Panos Constantopoulos[b]
Department of Informatics, Athens University of Economics and Business, Athens, Greece

Keywords: Ontology-Driven Information Extraction, Information Extraction from Text, Transformer-Based Methods, Relation Extraction, Knowledge Graph Creation.

Abstract: We present transformer-based methods for extracting information about research processes from scholarly publications. We developed a two-stage pipeline comprising a transformer-based text classifier that predicts whether a sentence contains the entities sought in tandem with a transformer-based entity recogniser for finding the boundaries of the entities inside the sentences that contain them. This is applied to extracting two different types of entities: i) research activities, representing the acts performed by researchers, which are entities of complex lexico-syntactic structure, and ii) research methods, representing the procedures used in performing research activities, which are named entities of variable length. We also developed a system that assigns semantic context to the extracted entities by: i) linking them according to the relation employs(Activity,Method) using a transformer-based binary classifier for relation extraction; ii) associating them with information extracted from publication metadata; and iii) encoding the contextualized information at the output into an RDF Knowledge Graph. The entire workflow is ontology-driven, based on Scholarly Ontology, specifically designed for documenting scholarly work. Our methods are trained and evaluated on a dataset comprising 12,626 sentences, manually annotated for the task at hand, and shown to surpass simpler transformer-based methods and baselines.

1 INTRODUCTION

Access to knowledge contained in research publications has become an increasingly demanding task, dramatically exaggerated due to the explosive growth rates of publications in scholarly domains (Bornmann et al., 2021). To address this problem, intelligent systems need to be able to capture the essence of knowledge represented in textual format, encode it according to concepts general enough to be applicable across disciplines and, at the same time, capable of representing semantic context that addresses the information needs of researchers, transforming the ways in which they engage with literature (Renear & Palmer, 2009). This type of encoded information can alleviate the task of keeping up to date in a specific domain, while maintaining a bird's-eye-view over a discipline or across disciplines, something particularly useful in interdisciplinary fields. To this end, generic concepts for representing knowledge of *"who has done what and how"* need to be modelled, appropriately identified in text and extracted into a knowledge base capable of answering questions of the form: *"find all papers that address a given problem"; "how was the problem solved"; "which methods are employed by whom";* etc. This goes beyond the retrieval features of search engines widely used by researchers, such as Google Scholar[1], Scopus[2] or Semantic Scholar[3] that mostly leverage bibliographic metadata, while knowledge expressed in the actual text is exploited mostly by matching query terms to documents.

Extracting and encoding the knowledge contained in research articles is a complex task which poses several challenges. For instance, the procedures employed by the researchers need to be identified and

[a] https://orcid.org/0000-0002-1007-4279
[b] https://orcid.org/0000-0001-6149-441X
[1] https://scholar.google.com/
[2] https://www.scopus.com/home.uri
[3] https://www.semanticscholar.org/

Pertsas, V. and Constantopoulos, P.
Ontology-Driven Extraction of Contextualized Information from Research Publications.
DOI: 10.5220/0012254100003598
In *Proceedings of the 15th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2023) - Volume 2: KEOD*, pages 108-118
ISBN: 978-989-758-671-2; ISSN: 2184-3228

extracted (as a kind of entities, here called "research methods") but also to be associated properly with their semantic context, i.e. the actual research activities in which they were used. To further capture the context of the research reported in an article (who is involved, what are their interests, affiliations, etc.), information must be drawn from the metadata of the article and associated with information extracted from the text, through mapping onto an appropriate schema.

In this paper we present Deep Learning (DL) methods for entity extraction along with a pipeline architecture based on a two-stage setup, consisting of an Entity Recogniser on top of a binary classifier, to extract two types of entities from the text of research articles: i) research methods, treated as named entities of variable length and ii) research activities, which appear as entities of complex lexico-syntactic structures. In addition, we present a system, that contextualizes the extracted entities by (1) extracting from the text the relation between research methods and research activities in which these are employed, using a transformer-based binary classifier we configured to this end; (2) by associating the information extracted from text with further contextual information derived from articles' metadata; and encodes all the derived knowledge into an RDF knowledge base adhering to Linked-Data standards. Finally, we present a manually created dataset consisting of 12,626 sentences specifically curated for the training and evaluation of our Information Extraction (IE) methods.

The rest of this paper proceeds as follows: in Section 2 we present background and related work; in Section 3 we describe the methodology and experimental setup; in Section 4 we report on the evaluation experiments; in Section 5 we discuss the evaluation results; and we conclude in Section 6 with insights for future work.

2 RELATED WORK

Information extraction (i.e. entity and relation extraction) from text constitutes an active research field where methodologies from ML and DL are employed and combined in different architectures and various domains. Entity extraction is usually treated as a token classification or sequence labelling task where a classifier predicts whether each token belongs to the entity in question or not. Relation extraction can be treated as text classification where the textual span representing the concatenation of specific extracted entities, or the text bounded somehow by those entities, is classified or not with the label of a designated relation among the entities. Through the years, various methods employing different techniques and designs have been proposed for dealing with information extraction from text. In works like (Chalkidis et al., 2017; Do et al., 2013; Pertsas et al., 2018), in order to extract entities and relations, the authors use a combination of engineered features representing textual, syntactic and surface form attributes along with various types of embeddings (word, POS, dependency, etc.) for vector representation, in tandem with ML models like SVM, Logistic Regression or CRFs that handle the classification task. In (Chiu & Nichols, 2016; Luan, Ostendorf, et al., 2018; Ma & Hovy, 2016; Peters et al., 2017) instead of handcrafted features, neural architectures like RNNs, LSTMS, BiLSTMS are employed for vector representation. In (Yu et al., 2020) the authors use ideas from graph-based dependency parsing to provide their model a global view on the input via a biaffine model (Dozat & Manning, 2017) and they show that it excels, especially in extracting nested entities, compared to CRF-based solutions. Similar conclusions are reached in (Lample et al., 2016), where transition-based parsers on top of Stack-LSTMs are considered as alternatives to CRFs for the sequence labeling task, with equal or higher performance. After the invention of large language models using transformer architectures (Vaswani et al., 2017), in more recent works like (He et al., 2022; Li et al., 2022; Nguyen & Huynh, 2022; Pandey et al., 2022; Pu et al., 2022) transformer-based language models like BERT or RoBERTa, that use self-attention to process input sequences and generate contextualized representations of words in a sentence, are employed either in combination with linear classifiers for text classification, or with CRFs in various architectures (like CRFs on top of ensembles of one-hot encoders) for the entity recognition (token classification) task.

In our methods, we combine transformer models with sigmoid activation layers on top for text classification. For entity extraction, inspired by (Lample et al., 2016), we use a transition-based parser for the sequence-labeling task, but this time combined with transformer models for text representation. In addition, we take one step further and experiment with a two-stage pipeline consisting of a transformer-based binary classifier for sentence classification and a transformer-, transition- based parser for boundary detection of the entities inside the sentence.

In parallel, systems that transform texts into Knowledge Graphs can be considered a useful component of the IE task, especially in the Science IE

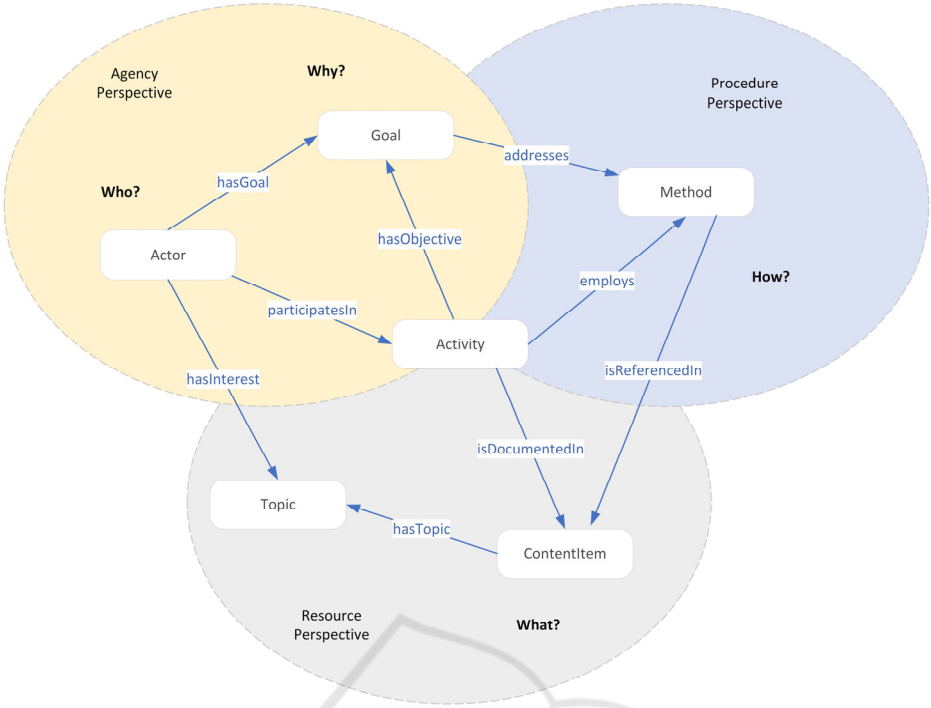

Figure 1: Scholarly Ontology core concepts and relations.

field where they have gained a lot of attention recently. Many efforts like (Dessì et al., 2020; D'Souza & Auer, 2022; Michel et al., 2020; Steenwinckel et al., 2020) use off-the-shelf solutions (such as DBpedia Spotlight, NCBO BioPortal Annotator, CrossRef, DyGIE++, etc.) in specialized domains of scientific literature, like Covid-19, Named Entity Recognition (NER) or Artificial Intelligence. Others like (Jaradeh et al., 2019) extract instances of generic concepts, namely those of process, method, material and data, using task-specific neural architectures atop pretrained SciBERT embeddings with a CRF-based sequence tag decoder. In (Luan, He, et al., 2018), the authors create a Knowledge Graph based on a unified learning model for extracting scientific entities, relations and coreference resolution. In our work we adopt a modular architecture allowing for easy future expansion where each module is independently trained for the task at hand. In addition, we train all our ML models on a dataset specifically curated for the task at hand. Furthermore, we use mechanisms for capturing semantic context by interrelating our extracted entities and by associating them with publication information and other Linked Data repositories like ORCID[6].

[6] https://orcid.org/

3 SETUP AND METHODOLOGY

3.1 Conceptual Model

The schema for the Knowledge Graph at the output as well as the definitions for all the concepts and relations we employ are provided by the Scholarly Ontology (SO) (Pertsas & Constantopoulos, 2017), a domain-independent ontology of scholarly/scientific work. A specialization, in fact precursor, of SO already applied to the domain of Digital Humanities is the NeDiMAH Methods Ontology (NeMO) (Constantopoulos et al., 2016). A brief overview of SO core concepts is given in the following section. For a full account see (Pertsas & Constantopoulos, 2017).

The core concepts and relations of SO, which form the basis for answering the "who", "what", and "how" in the scholarly domain, are displayed in Figure 1. The ontology is built around the central notion of *activity*, viewed from three different perspectives: i) the *agency* perspective, concerning actors and intentionality; ii) the *procedure* perspective, concerning the intellectual framework and organization of work; and iii) the *resource* perspective, concerning the material and immaterial objects consumed, used or produced in the course of activities.

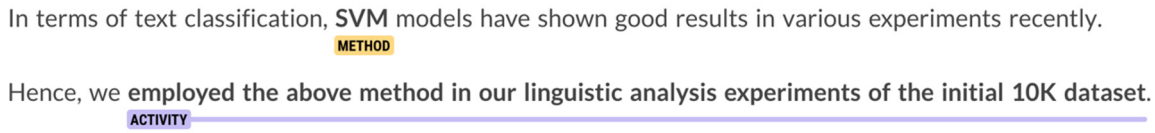

Figure 2: Textual spans of Method employed by Activity in neighbouring sentence and active voice.

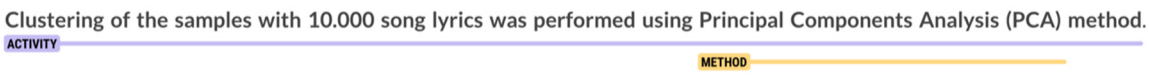

Figure 3: Textual spans of a Method, employed by an Activity in passive voice.

In this paper we are dealing with the extraction of textual representations of the SO concepts *Activity* and *Method* as well as the relation *employs(Act,Meth)* among them. Instances of the *Activity* class represent research processes or steps thereof such as an experiment, a medical or social study, an archaeological excavation, etc. They usually manifest in text as spans of phrases in passive or active voice in first person singular or plural, according to the number of authors who are their actual participants.

In contrast to activities, which are actual events carried out by actors, instances of the *Method* class denote procedures, such as an algorithm, a technique or a scheme that can be employed during an activity and describe how this was carried out. They are usually designated by single or multiple word terms, e.g. "ANOVA", "radio-carbon dating", etc., so their manifestations in text are mostly identified as named entities. Examples of textual manifestations of the classes *Activity*, *Method* and of the relation *employs(Act,Meth)* are shown in Figures 2 and 3.

SO provides for associating activities with their respective methods and participants through the relations *employs(Act,Meth),* and *hasParticipant(Act,Actor)* respectively. Provenance (e.g. the research article from where they were extracted) of those entities and relations can be modeled through the SO relations *isDocumentendIn(Act,ContentItem)* or *isReferencedIn(Meth,ContentItem)* and the class *ContentItem*, a subclass of the SO *Information Resource* Class which denotes all information resources (or parts thereof) utilized for representing content (e.g. articles, paragraphs, sections, etc.) independently of their physical carriers.

3.2 The Dataset

To train and evaluate all our ML models we used a manually annotated dataset consisting of 12,626 sentences. These sentences were derived from over 3,200 research articles (abstracts and main text) from 305 publishers spanning over 160 disciplines and research subfields so as to cover a broad variety of writing styles. The sentences were manually annotated by three human annotators. After appropriate training in SO, the annotators participated in 5 consecutive annotation trials covering in total 500 sentences from 300 papers. Each trial was followed by discussion on the results and evaluation of the Inter Annotator Agreement (IAA) using the Cohen's Kappa metric for IAA on individual couples and Fleiss' Kappa for the group IAA. After the trials, the best IAA scores reached 0.90 for *Activity*, 0.92 for *Method* and 0.91 for *employs(Act,Meth)* respectively yielding sufficient agreement levels so that annotators could subsequently work on separate datasets. The entire annotator training process lasted approximately 20 hours.

The annotation statistics of the final dataset after adjudication, are shown in Table 1. At sentence level the dataset contains 10,178 labels (i.e. each time a sentence contains an entity, it is assigned a respective label). At span level (as a span we consider each individual text-chunk that is annotated as an entity) there are in total 11,896 entity labels (i.e., labels assigned to spans to denote them as activities or methods). Finally at token level (as tokens we consider individual lexical units like words, punctuation marks, etc.) the dataset contains 128,914 labels (i.e., annotation labels assigned to tokens to denote them as part of a textual span representing an activity and/or a method). In addition, we created 4,754 textual spans bounded by any combination of the extracted activities and methods, even if they derive from neighbouring sentences (in order to capture any possible coreferences), out of which 2,284 were annotated as *employs(Act,Meth)*.

Compared to other published benchmarks in ScienceIE tasks (Augenstein et al., 2017; Jain et al., 2020; Luan, He, et al., 2018; QasemiZadeh & Schumann, 2016) our dataset shows similar or higher number of annotations, which renders it a good source for ground truth in such experiments. The dataset was randomly shuffled and split (hold-out method) into training, development, and evaluation sets with the

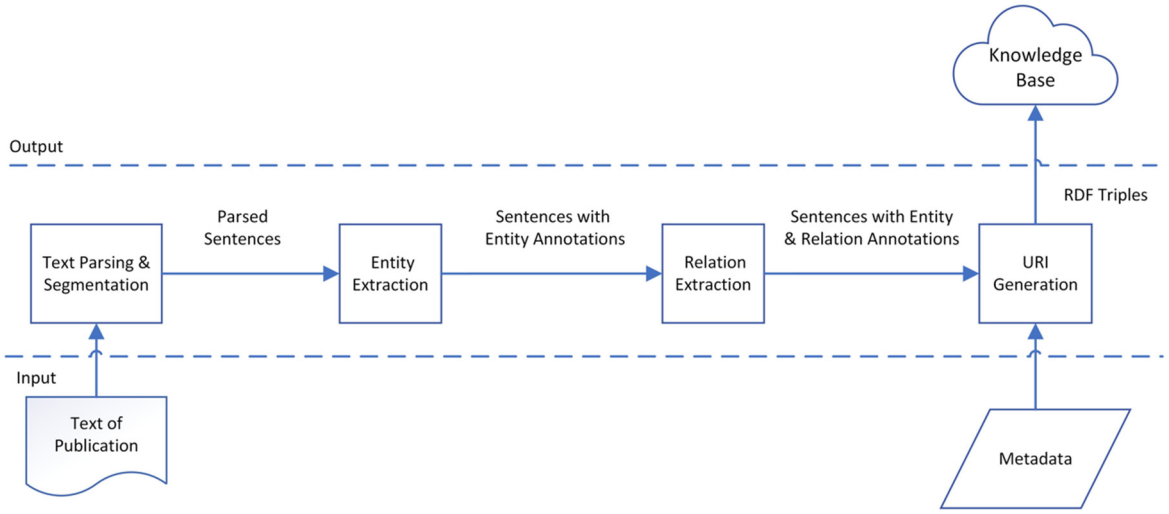

Figure 4: KG Creation. System architecture.

latter consisting of 20% of the entire dataset and the development set consisting of 10% of the rest.

Table 1: Dataset Characteristics.

Annotations	Activity	Method	Total
Sentence-level	5018	5160	10,178
Span-level	5506	6390	11,896
Token-level	110,298	18,616	128,914
employs(A,M)	2,284 spans		4,754 spans

3.3 Entity Extraction

For the entity extraction task, we developed a module that receives as input a sentence, tokenizes it and performs token-based classification in order to identify whether each token belongs to a textual span representing one of the entities to be extracted. Because our entities can be overlapping, we treat each entity recognition task independently. This also allows for further entity recognizers to be added easily in the future without having to retrain the previous ones. In the work reported here, we use two binary classifiers in order to identify and extract textual spans for *Activities* and *Methods*. Specifically, for each entity type we use two deep learning entity recognizers by combining Bert-base-NER and Roberta-base transformer models from the Hugging-Face library [7] for vector representation and a transition-based parser for the sequence labelling part. Both transformer models and the transition-based parser were fine-tuned / trained on the same dataset. These are the models **A-BERT-base-NER**, **A-RoBERTa-base**, for the extraction of Activities and **M-BERT-base-NER** and **M-RoBERTa-base** for the extraction of Methods.

In addition, we developed a two-stage pipeline consisting of a text (sentence) classifier with an entity recognizer on top (models: **A-Pipeline** and **M-Pipeline** for Activities and Methods respectively). The text classifier -consisting of a Transformer model (RoBERTa-base) and a sigmoid activation layer- receives as input a sentence and predicts whether it contains the requested entity or not. If yes, the sentence is then passed to the entity recognizer -consisting of a transformer with a transition-based parser- that performs token-based classification to identify the boundaries of the extracted entity. The entity recogniser (second component) of the pipeline is trained only in detecting the boundaries of entities inside sentences that contain them. The intuition behind the pipeline is that, by splitting the task into two simpler sub-tasks, each separate classifier will achieve high enough accuracy for their concatenation to produce better results despite any error propagation.

3.4 Relation Extraction

Extracting relations requires examining all plausible entity pairs. For every pair of extracted activity/method, the text chunk bounded by these two entities, [entity1, ..., entity2], is treated as expressing a candidate relation. In order to restrict the search to a reasonable set of candidates but also to allow for

[7] https://huggingface.co/models

```
<rdf:Description rdf:about="http://scholarly_ontology_instances#ACTIVITY/jstor.org/stable/symbinte.38.4.575/5/4_offset_0_40">
  <rdf:type rdf:resource="http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#String"/>
  <rdf:type rdf:resource="http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core#OffsetBasedString"/>
  <rdf:type rdf:resource="http://scholarly_ontology_schema#Activity"/>
  <rdf:type rdf:resource="http://scholarly_ontology_schema#Event"/>
  <rdf:type rdf:resource="http://scholarly_ontology_schema#SO_Entity"/>
  <rdfs:label>All individuals were assigned pseudonyms</rdfs:label>
  <ns2:referenceContext>http://www.jstor.org/stable/symbinte.38.4.575</ns2:referenceContext>
  <ns2:sentence>All individuals were assigned pseudonyms to ensure confidentiality.</ns2:sentence>
  <ns2:beginIndex rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">0</ns2:beginIndex>
  <ns2:endIndex rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">40</ns2:endIndex>
  <ns1:isDocumentedIn rdf:resource="http://scholarly_ontology_instances#ContentItem/jstor_http://www.jstor.org/stable/symbinte.38.4.575"/>
  <ns1:hasParticipant rdf:resource="http://scholarly_ontology_instances#Person/Sarah_Zelner"/>
  <ns1:hasGoal rdf:resource="http://scholarly_ontology_instances#GOAL/jstor.org/stable/symbinte.38.4.575/5/4_offset_44_66"/>
</rdf:Description>
```

Figure 5: Excerpt from produced RDF Triples.

possible coreferences among sentences to be resolved, a maximum limit of two (neighbouring) sentences is set for candidate creation. A classifier then determines whether the bounding entities of the chunk satisfy the property *employs(Act,Meth)*.

To this end, we developed a module that takes as input the candidate textual span and predicts whether the bounded entities satisfy the *employs(Act,Meth)* relation. For the classification, we used a binary classifier consisting of a transformer (BERT-base-uncased and RoBERTa-base) model for text representation, followed by a sigmoid activation layer (models **BERT-based-uncased** and **RoBERTa-base**). Both the transformer model and the classifier are fine-tuned/trained on the same dataset.

In addition, all the information extracted from text (entities and relations) is further related to publication information extracted from articles' metadata. The linking is based on post processing rules that map the extracted activities and methods to the researchers (authors of the paper) based on the corresponding SO properties of *hasParticipant* and *usesMethod* respectively. Provenance (e.g. the research article from where they were extracted) of those entities and relations can be modelled through the SO relations *isDocumentendIn* or *isReferencedIn* among the classes *Activity* / *Method* and the class *ContentItem* which denotes all information resources (or parts thereof) independently of their physical carriers. An overview of the entire system's modular architecture is given in Figure 4.

3.5 URI Generation

At the final stage we transform the extracted and contextualized information into RDF triples adhering to the Linked Data standards. Specifically, regarding the entities extracted from publication metadata, we produce URIs for the authors (further linked, when possible, with ORCID using the provided API) and their affiliations (as instances of the SO *Actor* class and corresponding subclasses *Person* and *Organization*), the research article (as instance of the *ContentItem* class) associated further with publication information, such as publisher, publication date, DOI etc., as well as disciplines associated with each publication and author keywords (as instances of the SO *Topic* class and its subclasses *Discipline* and *TopicKeyword* respectively), using the namespaces for SO, SKOS, RDFS and RDF when appropriate.

Regarding the entities extracted from text, we produce URIs for each extracted entity and relation using the NIF[8] and SO namespaces for preserving the context of the corresponding sentence and instantiating the appropriate classes and relations respectively. The output of the module is a set of RDF triples containing approximately 300 triples per publication on average. Figure 5 illustrates an example of the produced triples regarding an extracted textual span (label: "All individuals were assigned pseudonyms") representing an instance of the *Activity* class.

4 EVALUATION

The evaluation of Information Extraction methods involves comparing classifier results against a "gold standard" produced by human annotators. To this end, a confusion matrix is calculated based on the true positives (TP) -correctly classified predictions-, false positives (FP) -incorrectly classified predictions-, true negatives (TN) -correctly non-classified predictions and false negatives (FN) -incorrectly non-classified predictions. Performance scores are then

[8] https://persistence.uni-leipzig.org/nlp2rdf/

Table 2: Entity Extraction Evaluation Results.

Model	Token-based			Entity-based-partial			Entity-based-strict		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
A-Baseline	0.68	0.71	0.69	0.60	0.61	0.60	0.40	0.51	0.51
A-BERT-base-NER	0.86	0.87	0.86	0.79	0.80	0.80	0.72	0.73	0.72
A-RoBERTa-base	0.82	**0.91**	0.85	0.76	**0.84**	0.79	0.69	0.76	0.73
A-Pipeline	**0.86**	0.90	**0.88**	**0.80**	0.83	**0.81**	**0.73**	**0.77**	**0.75**
M-Baseline	0.62	0.66	0.65	0.66	0.73	0.70	0.55	0.60	0.57
M- BERT-base-NER	0.72	**0.89**	0.80	0.69	**0.85**	0.76	0.63	0.78	0.70
M-RoBERTa-base	0.75	0.85	0.78	0.73	0.83	0.77	0.67	**0.76**	0.70
M-Pipeline	**0.80**	0.83	**0.82**	**0.76**	0.79	**0.78**	**0.70**	0.72	**0.71**

Table 3: Pipeline components evaluation.

Pipeline	**P**	**R**	**F1**
A-TexCat	0.91	0.93	0.92
A-NER	0.80	0.82	0.81
A-Pipeline	0.86	0.90	0.88
M-TextCat	0.90	0.91	0.90
M-NER	0.78	0.79	0.79
M-Pipeline	0.80	0.83	0.82

Table 4: Relation Extraction evaluation.

Model	**P**	**R**	**F1**
Baseline	**0.83**	0.78	0.80
BERT-base-uncased	0.73	**0.95**	0.83
RoBERTa-base	0.82	**0.95**	**0.87**

measured based on Precision (P), Recall (R) and F1 scores computed as follows:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}, F1 = \frac{2 * P * R}{P + R}$$

For the entity extraction task, we conduct three types of evaluation experiments inspired by the guidelines in (Segura-Bedmar et al., 2013): 1) *token-based*, where a true positive (TP) is a token correctly classified as part of a chunk representing the entity, etc.; 2) *entity-based -partial matching*, where some overlap between the tagged entity and the "golden" entity is required but counts as half compared to the exact matches and 3) *entity-based -strict matching*, where only exact boundaries of the entities are counted for the match.

Table 2 shows evaluation results for each method and entity type. For comparison reasons we included a baseline model consisting of a residual CNN with bloom embeddings that utilize a stochastic approximation of traditional embeddings in order to provide unique vectors for a large number of words without explicitly storing a separate vector for each of them (Miranda et al., 2022), trained on the same dataset. Regarding the pipeline classifier which consists of a sentence- and a token-based classifier in tandem, detailed per stage and aggregate performance results are shown in Table 3. The aggregate scores of the pipeline are also shown in Table 2 for comparison with the other methods.

The evaluation of *employs(Act,Meth)* relation extraction methods involves comparing the predicted relations among the gold standard entities in each test set with those produced by the human annotators on the basis of Precision, Recall and F1 scores calculated as usual. A true positive (TP) is a chunk [entity1,...,entity2] for which the classifier correctly predicted the *employs(act, meth)* property; a false positive (FP) is a chunk for which *employs(Act,Meth)* was incorrectly predicted; and a false negative (FN) is a chunk for which *employs(Act,Meth)* incorrectly failed to be predicted. Table 4 shows the evaluation results for the *employs(Act,Meth)* relation extraction models using binary classifiers on top of Bert-base-uncased and RoBERTa language models as well as bloom embeddings (as baseline). All models are fine-tuned/trained on the same dataset.

5 DISCUSSION

As a general comment regarding the Information Extraction from text task, all methods except the baselines performed well, demonstrating the capabilities of transformer-based language models in text representation over the simpler neural-based methods (i.e. embeddings). Among the transformer models, the RoBERTa-base outperformed BERT-base in almost every metric, which can be attributed to the fact that the RoBERTa model is pre-trained on a much larger dataset (10 times larger) than BERT and uses a dynamic masking technique during training that helps it learn more robust and generalizable representations of words.

5.1 Entity Extraction Evaluation

Concerning the entity extraction methods, the pipeline architecture exhibited the best performance in all experiments and entity types, yielding up to 4% performance increase (F1 M-Pipeline) which proves our initial intuition that -splitting the task into two simpler ones -those of i) sentence classification and ii) boundary detection only in sentences with entities- would yield overall better performance, despite the possible error propagation.

The latter seems to be of less importance since the sentence classifier (first component) achieves very high recall, which yields the majority of the sentences containing entities to the input of the second component that focuses only on detecting the boundaries inside the sentence. However, as can be observed from the difference in performance among the various evaluation experiments (e.g. compare the F1 scores among token-based, entity-based-partial and entity-based-strict evaluations), the exact boundaries of the entity -even with the pipeline- are difficult to capture. This can be attributed to the fact that our entities are quite complex in terms of length variance, or/and lexico-syntactic structure which makes them difficult to isolate from the rest of the text in a sentence.

Regarding the *Method* class, lesser performance compared to the *Activity* classifiers can be attributed to the fewer labelled tokens on the dataset since each *Method* entity consists of much fewer tokens, on average. This could also be the reason for the lesser performance of the M-NER boundary detector model that "drags down" the entire M-pipeline performance. Error analysis showed that most errors were attributed to cases with long names (more than 3 tokens) and /or names containing multiple punctuation marks that seemed to confuse the classifier in identifying the correct boundaries of the entity (e.g. "Tests of population genetic structure and nested clade phylogeographic analysis (NCPA) [12,13] were used to infer connectivity on both sides of the Amazon basin."). However, even in such cases, at least 50% of the tokens of the span representing the entity name were captured correctly by the classifier.

Regarding the recognition of textual spans representing instances of the *Activity* class, error analysis attributes the majority of errors to cases of passive voice where the agent (implied or not) was other than the authors of the paper. These cases usually refer to statements regarding someone else (e.g. "In [54] linguistic analysis was performed on 10000 samples from MeCaB database.") and seem to be classified (erroneously) as research activities of the authors, although this is clearly not the case. However, similar cases of agents other than the authors of the paper, but in active voice, do not seem to confuse the classifier. This can be attributed to the lexico-syntactic complexities of passive voice, especially when the length of the sentence is big, and could probably be resolved with more training data (focused on passive voice).

Additional errors were detected in boundaries of the entities, where the classifier seemed to set the end-boundary one or two tokens after the proper end (e.g. "We collected data from more than 200 crania to test hypotheses about the relationships between cranial variation and genetic time and space in the Aleutian Islands."). In such cases, where only the last one or two tokens of the entity are erroneously classified, the boundary can be fixed with post processing rules.

5.2 Relation Extraction Evaluation

In relation extraction, like in entity extraction, transformer-based models exhibited superior performance. Error analysis suggests that misclassifications are mostly due to sentences with multiple entities, where the classifier failed to interrelate all the entities properly. For example, consider the excerpt: "Pre-test data were collected and analysed using **T-test**, while post-test were further analysed using **ANOVA** and **two-way ANOVA**." Here the classifier erroneously related the methods ANOVA and two-way ANOVA to the first activity ("pre-test data were collected and analysed using T-test"), although this is not the case based on the actual text.

In addition, in cases with adjacent sentences where both sentences contained a lot of entities, the majority of misclassifications involved a method of the first sentence being erroneously related to the second sentence even if this was clearly not the case. For example, consider the excerpt: "We employed both **Logistic Regression** and **SVM** to conduct the classification experiments and then analysed the results using **T-test**. The latter was used in combination with **ART tests** to prove the statistical significance of our results.". In this case the classifier erroneously related all the extracted methods (Logistic Regression, SVM and T-Test) of the first sentence, with the activity of the second sentence, although this was the case for only the last one (T-test).

Concerning the two-sentence limit for textual chunks construction, by visual observation of more than 1000 relations, we were able to notice that the majority of references occurred inside the sentence

with very few cases of reference to a method from an adjacent sentence and none more than two sentences apart, thus justifying the heuristic for a two-sentence limit in chunk creation.

5.3 Knowledge Graph Creation

Regarding the KG creation using post-processing rules that extract metadata and link them to the extracted entities, the system exhibited very good performance since it relies solely on pre-constructed mappings between fixed schemas. Few isolated incidents (lower than 1%) of improper association were due to errors in tags (e.g. non-Unicode characters or few missed entries that produced inconsistencies) and can be treated with additional escape rules as part of the general debugging process.

The KG created as described in this paper offers structured semantic views of the content of publications, which enhance our capability for comprehensive exploration of research work. This can be demonstrated through semantically complex queries executed over the KB. Indicative such queries, expressed in SPARQL are presented below:

Query 1: Retrieve all activities that employ any method regarding reconstruction (e.g. 2D or 3D reconstruction) in Paleontology.

```
SELECT DISTINCT ?a_label
WHERE {
 ?a rdfs:label ?a_label.
 ?a so:employs ?m.
 ?m so:comesFromDiscipline so:Paleontology.
 ?m rdfs:label ?m_label.
 filter
contains(lcase(?m_label),"reconstruction").}
```

Here, using the SO relation (*so:comesFromDiscipline*) and the filter *contains* SPARQL expression, all the method labels containing –in lower or upper case- the word "reconstruction" and coming from the discipline of Paleontology can be retrieved. They are then associated with the activities that employ them.

Query 2: For a specific paper (e.g. "Paper1"), retrieve all the research activities conducted by the authors along with the methods they employed.

```
SELECT ?m_label ?a_label
WHERE {
 ?a so:isDocumentedIn so:Paper1.
 ?a rdfs:label ?a_label.
 ?a so:employs / rdfs:label ?m_label.}
```

Here, through the use of property chains in SPARQL, the overall activity reported in a paper is decomposed into a series of activities denoting "what" the authors have done, associated with the methods employed during those activities. In this way, questions of "what" and "how" regarding the activities described in a research publication can be answered. Thus, the reader obtains an enhanced "bird's-eye" view of what is described in a paper before actually reading it. Additional information regarding the authors and their research interests, can also be retrieved using the appropriate SO classes and relations.

6 CONCLUSION

We presented a system that extracts contextualized information from scholarly publications and creates an RDF Knowledge Graph. The entire process is ontology-driven, based on the concepts and definitions provided by Scholarly Ontology, specifically designed for documenting scholarly work. We focused on the extraction of two types of entities, research methods and research activities, and of the relation *employs(Act,Meth)* denoting that an activity employs a particular method. This was integrated with information drawn from publication metadata, also on the basis of the ontology.

For the entity extraction task, we used transition-based parsers on top of BERT and RoBERTa transformer models for text representation. In addition, we used a two-stage pipeline architecture, comprising a transformer-based binary classifier for detecting the existence of entities in a sentence, in tandem with a transition-based parser for boundary detection.

To extract the *employs(Act,Meth)* relation, we used two transformer-based binary classifiers employing BERT and RoBERTa language models respectively. All our ML models are fine-tuned and trained on the same -manually curated- dataset consisting of 12,626 sentences from scholarly publications, specifically curated for the tasks at hand.

Evaluation was based on token-based, entity-partial-based and entity-exact-based calculations of P, R and F1 scores for each ML method and entity/relation type. Results showed higher performance of the pipeline architecture and overall good performance of the transformer-based models compared to a baseline approach that uses embeddings for textual representation.

Future work includes expanding the Knowledge Graph with recognition and extraction of other entities of the ontology, such as researchers'

assertions based on the outcomes of their activities, the activities' objectives (i.e., research goals) and information from citations, as well as linking extracted entities with other knowledge bases across the Web, such as Wikidata, for named entity mentions and Open Citations for bibliographic information.

ACKNOWLEDGEMENTS

The authors would like to thank Marialena Kasapaki and Panagiotis Leontaridis for their important contributions to the training of the DL models.

REFERENCES

Augenstein, I., Das, M., Riedel, S., Vikraman, L., & McCallum, A. (2017). SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 546–555. https://doi.org/10.18653/v1/S17-2091

Bornmann, L., Haunschild, R., & Mutz, R. (2021). Growth rates of modern science: A latent piecewise growth curve approach to model publication numbers from established and new literature databases. *Humanities and Social Sciences Communications*, *8*(1), 224. https://doi.org/10.1057/s41599-021-00903-w

Chalkidis, I., Androutsopoulos, I., & Michos, A. (2017). Extracting contract elements. *Proceedings of the 16th Edition of the International Conference on Articial Intelligence and Law*, 19–28. https://doi.org/10.1145/3086512.3086515

Chiu, J. P. C., & Nichols, E. (2016). Named Entity Recognition with Bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, *4*, 357–370. https://doi.org/10.1162/tacl_a_00104

Constantopoulos, P., Hughes, L. M., Dallas, C., Pertsas, V., & Christodoulou, T. (2016). Contextualized Integration of Digital Humanities Research: Using the NeMO Ontology of Digital Humanities Methods. *Digital Humanities 2016*, 161–163.

Dessì, D., Osborne, F., Reforgiato Recupero, D., Buscaldi, D., Motta, E., & Sack, H. (2020). AI-KG: An Automatically Generated Knowledge Graph of Artificial Intelligence. In J. Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne, & L. Kagal (Eds.), *The Semantic Web – ISWC 2020* (Vol. 12507, pp. 127–143). Springer International Publishing. https://doi.org/10.1007/978-3-030-62466-8_9

Do, H. H. N., Chandrasekaran, M. K., Cho, P. S., & Kan, M. Y. (2013). Extracting and matching authors and affiliations in scholarly documents. *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries - JCDL '13*, 219. https://doi.org/10.1145/2467696.2467703

Dozat, T., & Manning, C. D. (2017). Deep Biaffine Attention for Neural Dependency Parsing. *Conference Track Proceedings*. 5th International Conference on Learning Representations, ICLR, Toulon, France.

D'Souza, J., & Auer, S. (2022). Computer Science Named Entity Recognition in the Open Research Knowledge Graph. *From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries*, *13636*. https://doi.org/10.1007/978-3-031-21756-2_3

He, J., Uppal, A., N, M., Vignesh, S., Kumar, D., & Kumar Sarda, A. (2022). Infrrd.ai at SemEval-2022 Task 11: A system for named entity recognition using data augmentation, transformer-based sequence labeling model, and EnsembleCRF. *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 1501–1510. https://doi.org/10.18653/v1/2022.semeval-1.206

Jain, S., Van Zuylen, M., Hajishirzi, H., & Beltagy, I. (2020). SciREX: A Challenge Dataset for Document-Level Information Extraction. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7506–7516. https://doi.org/10.18653/v1/2020.acl-main.670

Jaradeh, M. Y., Oelen, A., Farfar, K. E., Prinz, M., D'Souza, J., Kismihók, G., Stocker, M., & Auer, S. (2019). Open Research Knowledge Graph: Next Generation Infrastructure for Semantic Scholarly Knowledge. *Proceedings of the 10th International Conference on Knowledge Capture*, 243–246. https://doi.org/10.1145/3360901.3364435

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural Architectures for Named Entity Recognition. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 260–270.

Li, Y., Yang, Y., Zhang, Y., & Xu, R. (2022). HITSZ-HLT at SemEval-2022 Task 10: A Span-Relation Extraction Framework for Structured Sentiment Analysis. *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 1406–1411. https://doi.org/10.18653/v1/2022.semeval-1.195

Luan, Y., He, L., Ostendorf, M., & Hajishirzi, H. (2018). Multi-Task Identification of Entities, Relations, and Coreference for Scientific Knowledge Graph Construction. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3219–3232. https://doi.org/10.18653/v1/D18-1360

Luan, Y., Ostendorf, M., & Hajishirzi, H. (2018). The UWNLP system at SemEval-2018 Task 7: Neural Relation Extraction Model with Selectively Incorporated Concept Embeddings. *Proceedings of The 12th International Workshop on Semantic Evaluation*, 788–792. https://doi.org/10.18653/v1/S18-1125

Ma, X., & Hovy, E. (2016). End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *Proceedings of*

the 54th Annual Meeting of the Association for Computational Linguistics, *1*, 1064–1074.

Michel, F., Gandon, F., Ah-Kane, V., Bobasheva, A., Cabrio, E., Corby, O., Gazzotti, R., Giboin, A., Marro, S., Mayer, T., Simon, M., Villata, S., & Winckler, M. (2020). Covid-on-the-Web: Knowledge Graph and Services to Advance COVID-19 Research. In J. Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne, & L. Kagal (Eds.), *The Semantic Web – ISWC 2020* (Vol. 12507, pp. 294–310). Springer International Publishing. https://doi.org/10.1007/978-3-030-62466-8_19

Miranda, L. J., Kádár, Á., Boyd, A., Van Landeghem, S., Søgaard, A., & Honnibal, M. (2022). *Multi hash embeddings in spaCy* (arXiv:2212.09255). arXiv. http://arxiv.org/abs/2212.09255

Nguyen, D., & Huynh, H. K. N. (2022). DANGNT-SGU at SemEval-2022 Task 11: Using Pre-trained Language Model for Complex Named Entity Recognition. *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 1483–1487. https://doi.org/10.18653/v1/2022.semeval-1.203

Pandey, A., Daw, S., & Pudi, V. (2022). Multilinguals at SemEval-2022 Task 11: Transformer Based Architecture for Complex NER. *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 1623–1629. https://doi.org/10.18653/v1/2022.semeval-1.224

Pertsas, V., & Constantopoulos, P. (2017). Scholarly Ontology: Modelling scholarly practices. *International Journal on Digital Libraries*, *18*(3), 173–190. https://doi.org/10.1007/s00799-016-0169-3

Pertsas, V., Constantopoulos, P., & Androutsopoulos, I. (2018). Ontology Driven Extraction of Research Processes. In D. Vrandečić, K. Bontcheva, M. C. Suárez-Figueroa, V. Presutti, I. Celino, M. Sabou, L.-A. Kaffee, & E. Simperl (Eds.), *The Semantic Web – ISWC 2018* (Vol. 11136, pp. 162–178). Springer International Publishing. https://doi.org/10.1007/978-3-030-00671-6_10

Peters, M., Ammar, W., Bhagavatula, C., & Power, R. (2017). Semi-supervised sequence tagging with bidirectional language models. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1756–1765. https://doi.org/10.18653/v1/P17-1161

Pu, K., Liu, H., Yang, Y., Ji, J., Lv, W., & He, Y. (2022). CMB AI Lab at SemEval-2022 Task 11: A Two-Stage Approach for Complex Named Entity Recognition via Span Boundary Detection and Span Classification. *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, 1603–1607. https://doi.org/10.18653/v1/2022.semeval-1.221

QasemiZadeh, B., & Schumann, A.-K. (2016). The ACL RD-TEC 2.0: A Language Resource for Evaluating Term Extraction and Entity Recognition Methods. *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 1862–1868.

Renear, A. H., & Palmer, C. L. (2009). Strategic Reading, Ontologies, and the Future of Scientific Publishing. *Science*, *325*(5942), 828–832. https://doi.org/10.1126/science.1157784

Segura-Bedmar, I., Martinez, P., & Zazo, M. H. (2013). SemEval-2013 Task 9: Extraction of Drug-Drug Interactions from Biomedical Texts (DDIExtraction 2013). *Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, *2*, 341–350.

Steenwinckel, B., Vandewiele, G., Rausch, I., Heyvaert, P., Taelman, R., Colpaert, P., Simoens, P., Dimou, A., De Turck, F., & Ongenae, F. (2020). Facilitating the Analysis of COVID-19 Literature Through a Knowledge Graph. In J. Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne, & L. Kagal (Eds.), *The Semantic Web – ISWC 2020* (Vol. 12507, pp. 344–357). Springer International Publishing. https://doi.org/10.1007/978-3-030-62466-8_22

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010.

Yu, J., Bohnet, B., & Poesio, M. (2020). Named Entity Recognition as Dependency Parsing. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6470–6476. https://doi.org/10.18653/v1/2020.acl-main.577